A Low Component Count Video Data Terminal Using the DP8350 CRT Controller and the INS8080 CPU

INTRODUCTION

The DP8350 is an I²L - LS technology integrated circuit, designed to provide all control signals for a cathode ray tube (CRT) display system. This application note explains a system using the DP8350 and the INS8080 microprocessor. The design philosophy shows how the DP8350 interfaces to the INS8080, completing the function of a video data terminal with a minimum component count. After reading and understanding this application note the reader will realize the ease and flexibility of designing video terminals with the DP8350*. To thoroughly understand this application note the reader must be familiar with the DP8350 and the INS8080 microprocessor.

The video data terminal described is divided into the following sections, (Figure 1).

The DP8350 CRT controller (CRTC).

The 8080 µP system which includes ROM, RAM, interrupt instruction port, oscillator, and control support chips.

The character generator.

The communication element.

The keyboard and baud rate select ports.

THE CRTC

The DP8350 generates all the required control and timing signals for displaying video information on the video monitor. Here is a summary of the controller's functions:

Dot clock, control, and counter outputs for the character generator.

Bidirectional RAM address refresh counter for refreshing the video RAM and allowing microprocessor loading to the internal DP8350 registers.

Direct drive horizontal and vertical sync signal outputs.

Direct cursor address location output. The cursor is internally delayed or pipelined, allowing for the access time of video RAM and the character generator ROM, (Figure 1).

THE CPU

The microprocessor provides CRTC, operator, and external machine control for the system. When the CRT controller is not actively refreshing the video RAM, (i.e., during vertical retrace or blank scan lines), the microprocessor is enabled for system housekeeping, (Figure 2). This method of multiplexing the RAM with *The DP8350 is equivalent to the INS8276

National Semiconductor Application Note 199 Al Brilliott August 1978



the CPU and the CRTC eliminates the need for line buffers.

THE CHARACTER GENERATOR

The character generator consists of 3 elements: an address latch to hold the input address to the character ROM allowing for the access time of the ROM; the character ROM that stores the ASCII character in a form for parallel to serial conversion by the shift register; the shift register converts the character ROMs parallel output to serial form. The serial output from the shift register is the true video output, modulating the video monitors electron beam which writes characters on the screen. All of the 3 elements of the character generator are combined in the DM8678, (Figure 3). The DP8350 CRTC provides all the control signals for the DM8678.

THE COMMUNICATION ELEMENT

The INS8250 is the asynchronous communication element (ACE) for the data terminal. The ACE allows the CPU portion of the data terminal communication with peripherals or host computers at the correct baud rate, (Figure 1). The ACE is programmed by the CPU to send and receive serial data at the standard baud rates from 110 to 4800 baud. The ACE, in conjunction with the DS1488 and DS1489 line drivers and receivers, also provides full RS232C synchronous communication if higher baud rates are desired. System communication speed must always be considered to insure the baud rate does not exceed the time required for the CPU to process a data byte. Asynchronous communication at baud rates higher than 4800 are possible by adding a line buffer.

SYSTEM INITIALIZATION

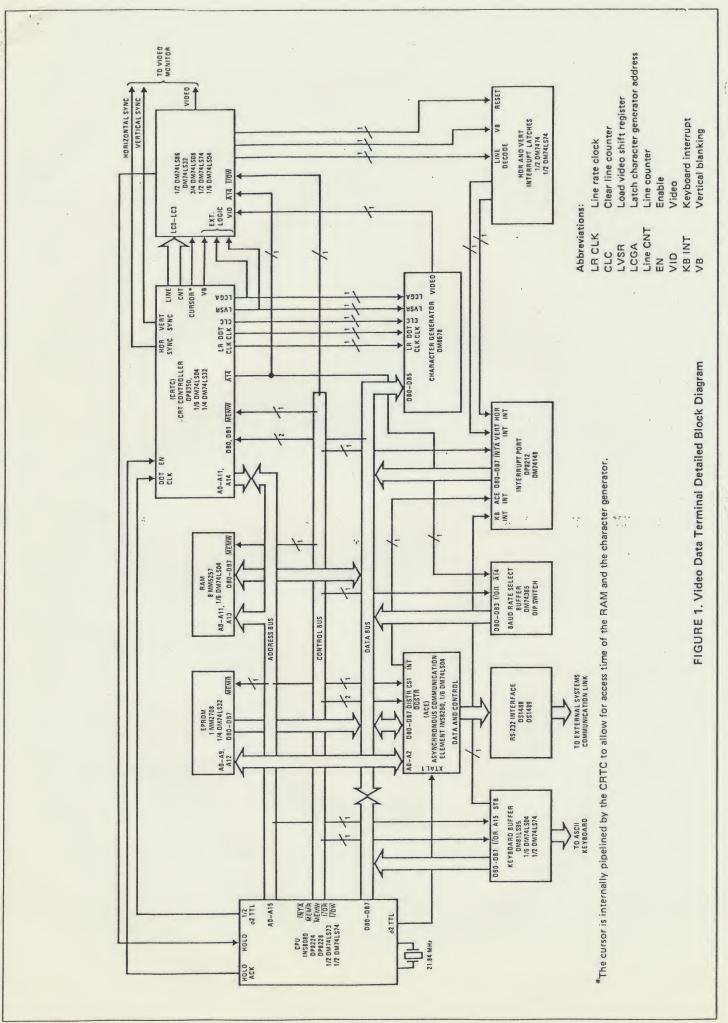
Application of the terminal's power supply resets the microprocessor, the communication element, and the CRT controller. Resetting the ACE is necessary to clear the interrupt. Resetting the CRTC is not absolutely necessary since the microprocessor loads the cursor and top of page registers in the initialization routine.

Following the reset all interrupts are disabled to avoid unwanted interrupts from the CRTC, ACE, or I/O ports. Refer to the initialization routine in the flowchart.

The stack pointer is loaded to the bottom of scratch pad RAM (3FFFH) for use as the register save pointer, (Figure 4). IFFF

The entire RAM is written with ASCII spaces generating a cleared screen. After completion of the screen clear loop the CPU writes 000H to the cursor and the top of page registers in the DP8350 CRTC. The routine homes the cursor to the upper left corner of the screen. The top of page register was loaded with 000H, therefore, the video RAM is refreshed by the CRTC from that starting address to the last address on the screen of video RAM (1920 characters).

...



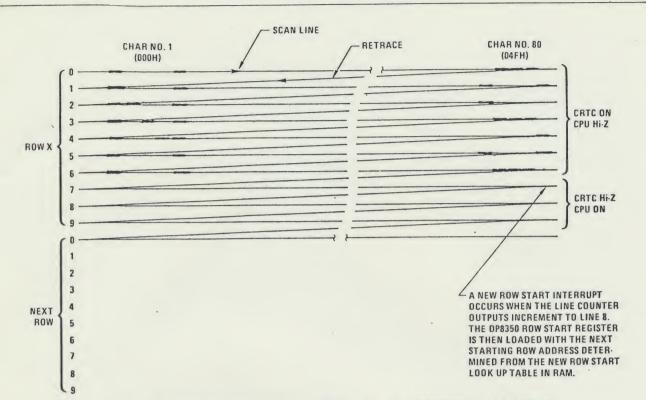


FIGURE 2. Row Start Interrupting and Multiplexing the INS8080 with the DP8350

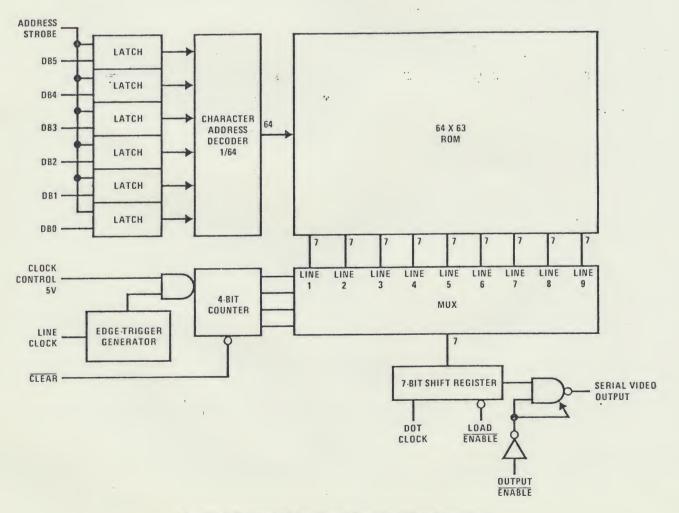


FIGURE 3. DM8678 Character Generator Block Diagram

The CPU is ready to perform the communication element (ACE) load routine. First, the baud rate divisor for the ACE must be determined. The baud rate select switch is read providing a code which corresponds to the appropriate 16-bit divisor for the ACE. This divisor determines the baud rate at which the ACE will communicate. Any additional programming requirements needed for the ACE to communicate with host computer systems could also be done at this time. The software in this system does not contain any additional programming for the ACE. There are many programming modes related to the ACE. Details of these modes are beyond the scope of this application note.

The row start look-up table, (Figure 5), is loaded up by a simple algorithm that loads and adds the data for referencing a row number to that row's starting address. The reference table, (Figure 6), is initialized next by direct loading. This table provides the CPU with top of page, bottom of page, next row load, cursor row, and scratch row numbers for system housekeeping.

Finally, the new row start and vertical interrupt latches are cleared, (Figure 7). The register pointers are loaded and the CPU is forced in a wait loop with interrupts enabled.

NON-SEQUENTIAL ADDRESSING

The data terminal described here was designed for non-sequential starting row addressing. In many systems sequential row addressing is used. If a character row consists of 10 scan lines the RAM is addressed 10 repetitive times from 000H through 04FH, (Figure 2). The next row is refreshed in the same manner from 050H to 09FH. The starting row address is sequential 7000H, 050H, 0A0H—EB0H for row numbers 0H, 1H, 2H,—2FH, respectively. Non-sequential row addressing would be equivalent to 050H, 000H, 0A0H—EB0H for row numbers 1H, 0H,—2FH, respectively, (Figure 4).

In conjunction with the CPU, non-sequential row addressing is quite easily accomplished with the DP8350 since this is one of the features designed into the part. Accomplishing this task basically requires the following sequence of events. Assume the CRTC has finished writing a video row in the middle of the monitor's screen. This system has a 5×7 character font in a 7×10 field, (Figure 2). At the completion of the last video scan line 7 the CRTC line counters continue to count the last 3 lines. Video is not present since the character is only 7 scan lines high. The blank scan lines are 7, 8, and 9 permitting the CRTC address outputs to be TRI-STATED®, allowing the CPU to run. When the line counter outputs increment to scan line 8 an interrupt signals the CPU. The interrupt occurring is the new row start interrupt. The interrupt routine fetches the next CRTC row number from the reference table (Figure 6). This number is converted to the new starting row address, explained later, and loaded to the CRTC row start register. The CPU finishes the routine by clearing the interrupt, readying itself for the next new row start interrupt. The entire routine takes 1 scan line of time, approximately 64 µs. The CRTC continues to scan the video RAM from that new starting address on for the next 7 repetitive scan lines of the next row. Many advantages become apparent using the nonsequential addressing scheme. Scrolling up or down with the cursor always on the screen may be done

faster and easier from a hardware/software standpoint. Exchanging one row with another row is fast since it is not necessary to rewrite the video RAM. Row swapping is useful for higher end terminals requiring row editing functions.

ADDRESS MAP

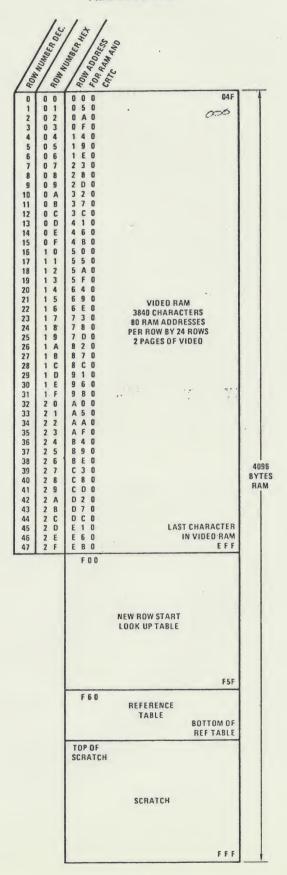


FIGURE 4. RAM Organization

Page 2

Page 1														
RC	W			N	RS	НІ	GH			N	RS	LC	W	
NUM	-	R X	ADDRESS ROW DATA			AE	ADDRESS				W TA			
0	0	0	3	F	0	0	3	0	3	F	3	0	0	0
1	0	1	3	F	0	1	3	0	3	F	3	1	5	0
2	0	2	3	F	0	2	3	0	3	F	3	2	Α	0
3	0	3	3	F	0	3	3	0	3	F	3	3	F	0
4	0	4	3	F	0	4	3	1	3	F	3	4	4	0
5	0	5	3	F	0	5	3	1	3	F	3	5	9	0
6	0	6	3	F	0	6	3	1	3	F	3	6	E	0
7	0	7	3	F	0	7	3	2	3	F	3	7	3	0
8	0	8	3	F	0	8	3	2	3	F	3	8	8	0
9	0	9	3	F	0	9	3	2	3	F	3	9	D	0
10	0	Α	3	F	0	A	3	3	3	F	3	Α	2	0
11	0	В	3	F	0	В	3	3	3	F	3	В	7	0
12	0	C	3	F	0	С	3	3	3	F	3	С	С	0
13	0	D	3	F	0	D	3	4	3	F	3	D	1	0
14	0	E	3	F	0	E	3	4	3	F	3	E	6	0
15	0	F	3	F	0	F	3	4	3	F	3	F	В	0
16	1	0	3	F	1	0	3	5	3	F	4	0	0	0
17	1	1	3	F	1	1	3	5	3	F	4	1	5	0
18	1	2	3	F	1	2	3	5	3	F	4	2	A	0
19	1	3	3	F	1	3	3	5	3	F	4	3	F	0
20	1	4	3	F	1	4	3	6	3	F	4	4	4	0
21	1	5	3	F	1	5	3	6	3	F	4	5	9	0
22	1	6	3	F	1	6	3	6	3	F	4	6	E	0
23	1	7	3	F	1	7	3	7	3	F	4	7	3	0

RC	ROW NRS HIGH						GH		NRS LOW					
NUN	BE		AE	DF	RES	ss	RC		AD	DF	RES	SS	RO	
24	1	8	3	F	1	8	3	7	3	F	4	8	8	0
		9	3	F	1	9	3	7	3	F	4	9	D	0
25	1		_				3		3	F	4	A	2	0
26	1	A	3	F	1	A		8	-					
27	1	В	3	F	1	В	3	8	3	F	4	В	7	0
28	1	С	3	F	1	С	3	8	3	F	4	С	C	0
29	1	D	3	F	1	D	3	9	3	F	4	D	1	0
30	1	E	3	F	1	E	3	9	3	F	4	E	6	0
31	1	F	3	F	1	F	3	9	3	F	4	F	В	0
32	2	0	3	F	2	0	3	Α	3	F	5	0	0	0
33	2	1	3	F	2	1	3	Α	3	F	5	1	5	0
34	2	2	3	F	2	2	3	Α	3	F	5	2	A	0
35	2	3	3	F	2	3	3	Α	3	F	5	3	F	0
36	2	4	3	F	2	4	3	В	3	F	5	4	4	0
37	2	5	3	F	2	5	3	В	3	F	5	5	9	0
38	2	6	3	F	2	6	3	В	3	F	5	6	E	0
39	2	7	3	F	2	7	3	С	3	F	5	7	3	0
40	2	8	3	F	2	8	3	C	3	F	5	8	8	0
.41	2	9	3	F	2	9	3	C	3	F	5	9	D	0
42	2	A.	3	F	2	Α	3	D	3	F	5	A	2	0
43	2	В	3	F	2	В	3	D	3	F	5	В	7	0
44	2	С	3	F	2	С	3	D	3	F	5	C	C	0
45	2	D	3	F	2	D	3	E	3	F	5	D	1	0
46	2	E	3	F	2	E	3	E	3	F	5	E	6	0
47	2	F	3	F	2	F	3	E	3	F	5	F	В	0

FIGURE 5. New Row Start Look Up Table

ı				
	FUNCTION	ADDRESS	DATA	INITIALIZED DATA
	Last Row #	3F60	XY	17
	8080 Row #	3F61	XY	00
	First Row #	3F62	XY	00
	Character #	3F63	XY	00
	CRTC Row #	3F64	XY	00
	Row Save #	3F65	XY	00
	Temp. 1	3F66	XY	00
	Temp. 2	3F67	XY	00

COMM	IAND	FUNCTION
OUT	40	Clear new row start and vertical interrupt latches
IN	80	Read keyboard
IN	40	Read baud rate select switch

FIGURE 7. Input/Output Space

FIGURE 6. Reference Table

DEVICE	ADDRESS*
ROM	0000 to 0FFF
RAM	3000 to 3FFF
CRTC	5000 to 5FFF
ACE	9000 to 9007

^{*}Direct device selecting was used to minimize the system component count

FIGURE 8. CPU Addressing Space

Γ	ROW NUMBER			NRS HIGH				NRS LOW								
					ADDRESS		ROW		ADDRESS				ROW			
1	DEC	Н	EX	AI	ADDRESS			DA	DATA		ADDRESS				DATA	
Г	32	2	0	3	F	2	0	3	Α	3	F	5	0	0	0	

Row Start Address for Row 20H. 3XXX Selects RAM.

5XXX Selects CRTC.

FIGURE 9. Example From the New Row Start Look Up Table

ROW LOADING DETAILS

Obtaining the next starting row address for the CRT controller is accomplished by an addressing and adding scheme from the new row start look-up table. The same scheme is used to determine any needed address, given the row number.

Figure 9 shows a row number and address taken from the new row start look-up table.

The row number is loaded from the reference table in RAM to a register. The CPU determines the starting address from the row number and stores it in a 16-bit pointer register. The higher order 4 bits contain address for the RAM or the CRT controller, (Figure 8).

Here are the details of how this is accomplished. Refer to the new row start interrupt in the software listing and *Figure 9*.

The CPU D-E registers are loaded to point to a row number in the reference table. The number is put in the accumulator and moved into the E register. The D-E register in this example now contains 3F20 which points to NRS HIGH ROW DATA (3A). The addressed data is moved to the accumulator and then to the H register. If it was desired to point to the CRTC then 20H would have been added to it first. The D-E register still contains 3F20H. To obtain the NRS LOW ROW DATA the E register is moved to the accumulator and 30H is added to it. Now the D-E register contains 3F50H and points to NRS LOW ROW DATA (00H). The data is loaded to the accumulator and then to the L register. The H-L registers contain 3A00H which is the starting row address for row number 20H. The method just described is used throughout the terminals program to move the cursor, load the top of page, and load the new starting row address in the CRTC.

VERTICAL INTERRUPT

The vertical interrupt occurs when the CRTC has completed refreshing a video page (1920 characters) of information. Vertical blanking identifies that condition and interrupts the CPU forcing it to the vertical interrupt routine. Refer to the vertical interrupt in the flow chart. The routine moves the first row number to the CRTC row number, updating it so the next new row start load occurs with the top of the page address or the first row of the video screen.

KEYBOARD INTERRUPT

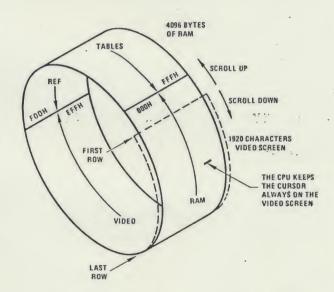
The external keyboard requirements are ASCII outputs with a suitable strobe to interrupt the CPU for keyboard servicing. Refer to the keyboard interrupt in the flow chart. After the keyboard buffer is read the data byte is tested for a (CNTL E), new baud rate command. If the test fails the CPU writes the data byte to the ACE. Passing the test forces the CPU to read the baud switch and load the ACE with the new baud rate.

ACE INTERRUPT

As mentioned above, a data byte read from the keyboard that is not a baud rate command enters the accumulator. The CPU writes the data byte from the accumulator to

the transmitter holding register in the ACE. The ACE proceeds to shift out the data byte, with the appropriate start and stop bits, serially from the (SOUT) output. The data is shifted to the serial input (SIN) of the ACE and loaded into the receiver holding register. When the register is full the ACE interrupts the CPU, initializating the ACE service routine. Refer to the ACE interrupt in the flow chart.

The CPU reads the receiver holding register in the ACE. Reading the ACE resets the interrupt. The data byte now resides in the accumulator. The CPU tests for a control or an escape function. The function is executed if test conditions are met. Refer to the keyboard interrupt routine in the software listing. The data byte is written to the video RAM at the cursor address which appears on the monitor screen. The cursor and character numbers are incremented as long as it is not at the end of a row. A character at the end of a row requires further testing to recognize the following situations. Is it the last row on the monitor's screen? Or is it on the maximum row of the video RAM? Essentially, the cursor is forced to stay visible on the video monitor's screen and video RAM is always kept out of scratch pad RAM. (Figure 10).



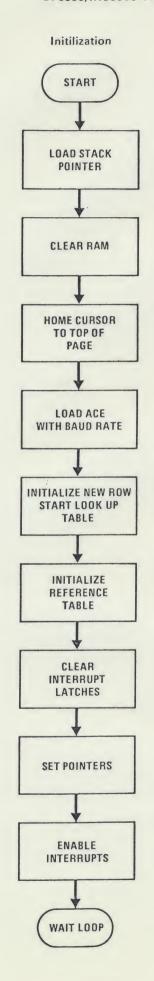
The video screen is allowed to scroll only through the video RAM (000H to EFFH). The CPU keeps the video screen within these bounds by loading the new row start register with that address range only (row 00H to 2FH).

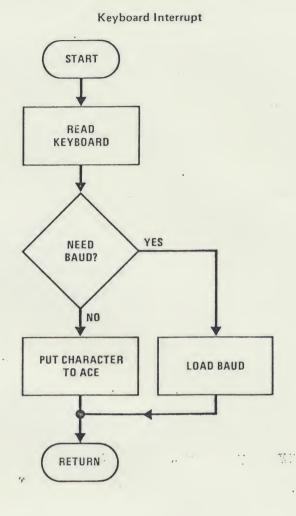
FIGURE 10. Drum Analogy for the RAM

FULL/HALF DUPLEX OPERATION

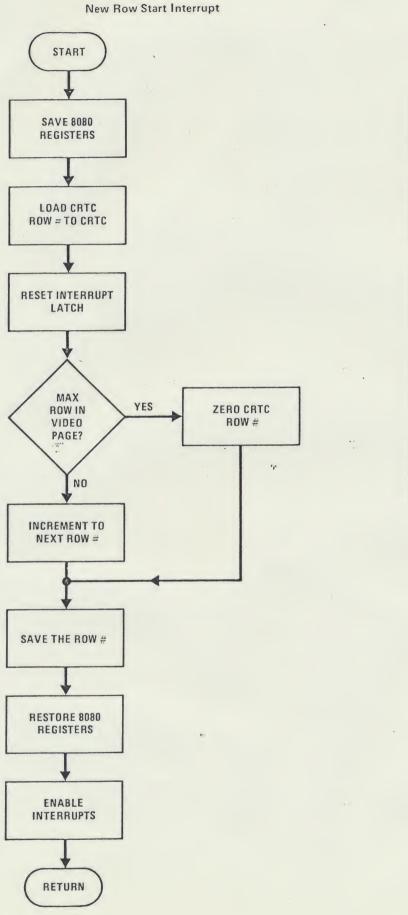
The data terminal and a host computer in the full duplex mode of operation would receive the serial information, process it, and send it back to the SIN input of ACE. Using the terminal in a stand-alone mode for testing, the serial out SOUT is tied to the serial in SIN of the ACE. In the half duplex mode a data byte is sent to the host computer at the same time it is sent to the terminal. When the data terminal is set up to communicate with a host computer the full duplex mode of operation is desirable.

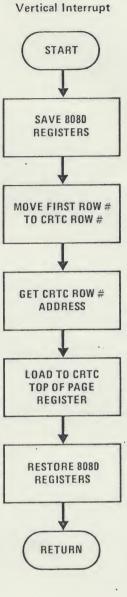
DP8350/INS8080 VIDEO DATA TERMINAL BASIC SOFTWARE FLOW CHART

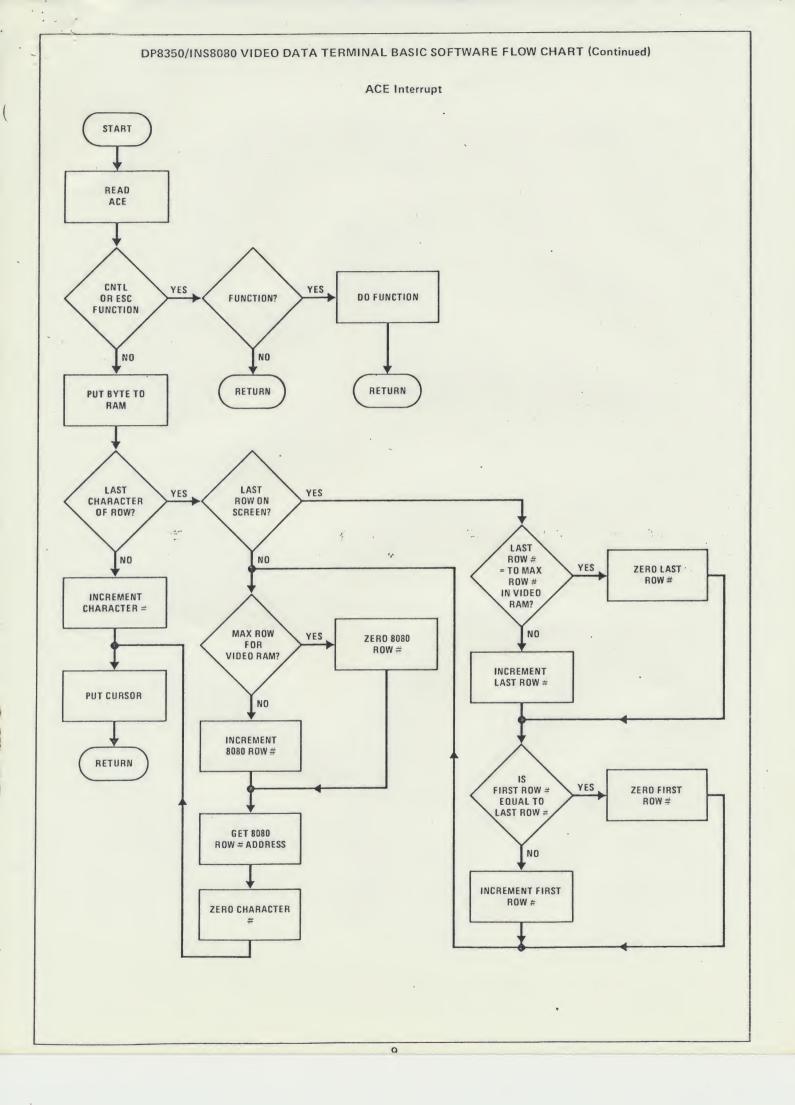




DP8350/INS8080 VIDEO DATA TERMINAL BASIC SOFTWARE FLOW CHART (Continued)







1 '	TITLE CETC	20304 6345,32	135 00E9 C31C01	.IMF	ACELD	
2 %		. 8030A 02/15/78	136 00EC 117E00 B1200 137 00EF C21C01	LXI	D. OOUTE ACELD	1200 BAUD DIVISOR
4 , 51	TIONAL SEMICONI RIES PROGRAMMAE	FLE CRT CONTROLLER BOARD	138 00F2 115400 E1800	LXI	D. 00054	; 1800 DAUD DIVISOR
6 , AL BI	HELIOTT-JIM TRO	OUTNER	139 00F5 031001 140 00F8 114000 B2000	LXI	D. 0004C	, 2000 BAUD DIVISOR
7 8 0060	LASTROW =	050	141 00FB C31C01 142 00FE 113F00 B2400	JMP LXI	ACELD D. 0003F	. 2400 BAUD DIVISOR
10 0062	ROWSOSO = FIRSTRO =	064 062	143 0101 031001 144 0104 112400 B3600	LXI	ACELD P. 0002A	3600 BAUD DIVISOR
11 006-3	CHARNIM =	063	145 0107 C31C01	JMF	ACELD	
12 0064 13 0065	CRICROW = ROWSAVE =	064	145 010A 112000 P4800 147 010D 031001	LXI JMP	D. 00020 ACELD	, 4800 BAUD DIVISOR
14 0066 15 0067	TEMP1 =	066	143 0110 111500 D7200 149 0113 C31C01	LX1	D. 00015 ACELD	.7200 BAUD DIVISOR
16 0069	IMASE =	068	150 0116 111000 E9600 151 0119 C31C01	LXI	D. 00010 ACELD	: 9600 BAUD DIVISOR
17 18 0000	#0000		152 -			
19 0000 F3 START 20 0001 31FF3F	DI LXI SF. 03	DISABLE INTERRUPTS LOAD STACE FOINTER	153 154		OAD ROUTINE	
21 0004 033800 22 0007	UMP INIT	JUMP TO INITILIZE ROUTINE	155 0110 010390 ACELD	LXI	B. 09003 A. 083	; POINT B C TO ACE ; INIT BAUD LOAD - 8 BITS
23 0008 €32502 24 000B	JMP NEWRO	O , NEW ROW START INTERRUFT	157 0121 02 158 0122 0E01	STAX	B C. 601	POINT TO BAUD HIGH
25 0010 C34A01	JMP INTAC	CE , ACE INTERUPT	159 0124 7A	MOV	A. D	GET BAUD HIGH STORE BAUD HIGH TO ACE
26 0013 27 0018 C33601	=0018 JMP INTLE	B . FEYBOARD INTERUPT	160 0125 02 161 0126 0E00	MVI	E C. 000	POINT ACE TO BAUD LOW
28 001B 29 0038 C34F02	#0038 JMP VERTI	I VERTICAL INTERUPT	162 0128 7B 163 0129 02	STAX	A, E B	STORE BAUD LOW TO ACE
30 003R 210000 INIT 31 003E 0E20	HVI C. 020	O , IST RAM ADDRESS O , ASCII SPACE INTO C REG	164 012A 0E03 165 012C 79	MOV	C, 003	; RESET DLAB TO ZERO ; INIT ACE T/R
32 0040 3E3F 33 0043 71 CLRAM	MVI A. CRE		166 0120 02	STAX	B C. 001	INTERRUPT ENABLE REG
34 0043 23	INX H	NEXT RAM ADDRESS	167 012E 0E01 168 0130 79	MOV	A. C	SELECT RECEIVED DATA INTERM"
35 0044 BC 36 0045 C24200	CMF H JNZ CLRAM	M , IF NO THEN NEXT ADD	169 0131 02 170 0132 0E00	MVI	C. 000	, LOAD IT ; RESTORE B-C ACE POINTER
37 0049 0E00 36 004A 3E40	MVI C. 000 MVI A. 040		171 0134 D1 172 0135 C9	POP	D	RESTORE D-E REGISTERS
39 0040 71 CLRAM: 40 004D 23			173		ARD INTERRUPT ROL	
41 004E BC	CMP H		175			, READ KEYROARD
42 004F C24C00 43 0052 CD8700	CALL HMCUR	R . GO TO CUR HOME FOUTINE	176 0136 DDS0 INTER 177 0138 FB	EI	080	ENABLE INTERRUPTS
44 0055 CD9300 45	CALL BAUD	GO TO BAUD LOAD ROUTINE	178 0139 FE05 179 013B CA9300	JZ JZ	005 BAUD	; NEED BAUD RATE? (CNTL E) ; IF YES GO TO BAUD ROUTINE
45	NEW ROW STAR	RT LOOF UP TABLE GENERATION	180 013E FE12 181 0140 CA4803	CPI	012 IVERTN	INVERT NEXT CNTL R
49 0058 21003F	LXI H. 03F		182 0143 FE13	CPI	013	INVERT ROW CNTL S
49 0058 11303F 50 005E 010030	LXI D, 03F	000 ; N R. S. ADDRESS DATA	183 0145 CA5403 184 0148 02	STAX	IVERTR B	STORE BYTE TO ACE
51 0061 70 NRS 52 0062 79	MOV M, B	, STORE TO N R S DATA TABLE , N. R. S. DATA LOW TO ACC.	186	RET		RETURN
53 0063 12 54 0064 0650	STAX D ADI 050	STORE TO N.R.S. DATA TABLE ACC READY FOR NEXT LOAD	L 187	; ACE II	NTERUPT ROUTINE	
55 0066 4F	MOV C.A	ACC TO N. R. S DATA HIGH	189 014A OA INTACE	LDAX	В	LOAD ACE DATA BYTE TO ACC.
56 0067 78 . 57 0068 CE00	ACI 000	, ADD CARRY BIT TO DATA HIGH		CPI	07E	
58 006A 47 59 006B 2C	MOV B.A	, MOVE RESULT TO N R S. DATA ; INCREMENT N R S. HIGH ADD	H 192 014E CA7001 193 0151 FE7F .	CPI	FUNC 07F	: TEST FOR ESC COMAMD : TEST FOR DEL COMAND
60 006C 1C 61 006D 7B	INR E MOV A.E	INCREMENT N.R.S. LOW ADD IN.R.S. ADD LOW TO ACC	194 0153 CA7001 195 0156 5F	JZ	FUNC E. A	SAVE CHAR IN REG E
62 006E FE60	CFI LASTE		196 0157 E660	ANI	060	MASK OUT BITS FOR CATL TEST
63 0070 C26100 64			197 0159 CA7001 198 015C 3A683F	JZ LDA	FUNC 03F68	LOAD INVERT MASK
65	REFERENCE TA	ABLE INITILIZE	199 015F B3 200 0160 77	ORA MOV	E M. A	STORE DATA BYTE TO RAM
67 0073 3E17 68 0075 12	MVI A. 017 STAX: D	7 ; LAST ROW NUMBER TO ACC , STORE TO REFERENCE TABLE	201 202	; ADIVANO	CE CURSOR .	. 4
69		HERAL INTERRUPT FLORS	203 204 0161 1E63 ADCUR	MUI	E, CHARNUM	FOINT B-C TO CHAR *
70		HERAL INITAROFI FLORS			CA CLIMETANDA	
71			" 205 0163 1A	LDAX	D	, LOAD CHAR # TO ACC
	OUT 040 IN 080	IN R S INTERRUPT CLEAR	205 0163 1A 206 0164 23 207 0165 FE4F		H O4F	NEXT CHAR LOCATION LAST CHAR OF ROW?
71 72 0076 0340 73 0078 0880 74	OUT 040 IN 080	IN R S INTERRUPT CLEAR KEYBOARD INTERRUPT CLEAR	φ 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01	LDAX INX CPI JZ	H O4F NXRO	NEXT CHAR LOCATION LAST CHAR OF ROW? LIF TRUE JUMP TO NEXT ROW
71 72 0076 £340 73 0078 £50 74 75 76	OUT 040 IN 080 SET UP FOINT	IN R.S. INTERRUPT CLEAR REYEGARD INTERRUPT CLEAR TERS	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12	LDAX INX CPI JZ ADI STAX	H 04F NXR0 001 D	, NEXT CHAR LOCATION , LAST CHAR OF ROW? ; IF TRUE JUMP TO NEXT ROW ; INCREMENT CHAR & ; STORE CHAP # TO RAM REF.
71 72 0076 £340 73 0078 £80 74 75 76 77 007A 11603F 78 007E 210030	OUT 040 IN 080 ,SET UP FOINT LXI D.03F LXI H.030	IN R S INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS F60 : POINT D-E TO REFERENCE TAI 000 : POINT H-L TO 1ST RAM LOCA	\$\psi\$ 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CARE01 209 016A C601 210 016C 12 LE 211 016D C3R301 1^2 212	LDAX INX CPI JZ ADI STAX JMP	H 04F NXRO 001 D PCUR	.NEXT CHAR LOCATION .LAST CHAR OF ROW? :IF TRUE JUMP TO NEXT ROW .INCREMENT CHAR &
71 72 0076 B340 73 0078 B880 74 75 76 77 0078 11603F 72 0076 210030 79 0080 010090 80	OUT 040 IN 080 ;SET UP FOINT LXI D.03F LXI H.030 LXI B.090	IN R S INTERRUPT CLEAR INEYDOARD INTERRUPT CLEAR TERS F60 IPOINT D-E TO REFERENCE TAI 000 IPOINT H-L TO 1ST RAM LOCA' 000 IPOINT B-C TO ACE	\$\psi\$ 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CARE01 209 016A C601 210 016C 12 LE 211 016D C3R301 1^2 212 213 214	LDAX INX CPI JZ ADI STAX JMP ; TEST F	H 04F NXRO 001 D PCUR FOR FUNCTION	, NEXT CHAR LOCATION , LAST CHAR OF ROW? ; IF TRUE JUMP TO NEXT ROW ; INCREMENT CHAR & ; STORE CHAP # TO RAM REF.
71 72 0076 0340 73 0078 DES0 74 75 76 77 0078 11603F 78 0076 210030 79 0080 010090	OUT 040 IN 080 ,SET UP FOINT LXI D.03F LXI H.030	IN R S INTERRUPT CLEAR INEYBOARD INTERRUPT CLEAR TERS F60 F0INT D-E TO REFERENCE TAI D00 F0INT H-L TO 1ST RAM LOCAT D00 F0INT B-C TO ACE OR INTERUPTS	\$\psi\$ 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213	LDAX INX CPI JZ ADI STAX JMP	H 04F NXRO 001 D PCUR	, NEXT CHAR LOCATION , LAST CHAR OF ROW? ; IF TRUE JUMP TO NEXT ROW ; INCREMENT CHAR & ; STORE CHAP # TO RAM REF.
71 72 0076 0340 73 0078 0880 74 75 75 76 77 007A 11603F 78 007D 210030 79 0080 016090 801 82 83 0083 FE BACK:	OUT 040 IN 080 ;SET UP FOINT LXI D.03F LXI H.03; LXI B.090 ;WAIT LOOP FO	IN R S INTERRUPT CLEAR KEYBOARD INTERRUPT CLEAR TERS FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO IST RAM LOCA' DOO FOINT B-C TO ACE OR INTERUPTS REMBELE INTERRUPTS	\$\psi\$ 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 212 213 214 215 0170 7B FUNC-216 0171 FE01 217 0173 CAD0000	LDAX INX CPI JZ ADI STAX JMP : TEST F	H O4F NXRO 001 D PCUR FOR FUNCTION A.E 001 START	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR W . STORE CHAP W TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM)
71 72 0076 0340 73 0078 DESO 74 75 75 76 77 007A 11603F 75 007D 210030 79 0050 010090 80 81 82 83 0083 FB BACK: 84 0084 C38300 85	OUT 040 IN 080 ;SET UP FOINT LXI D.095 LXI H.031 LXI E.09: ;WAIT LOOP FO	IN R S INTERRUPT CLEAR KEYBOARD INTERRUPT CLEAR TERS FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO IST RAM LOCA' OR INTERUPTS FOR LOCAL INTERRUPTS LOOP UNTIL INTERRUPTED	\$\phi\$ 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC-216 0171 FE01 217 0173 CA0060 218 0178 C46E02	LDAX INX CPI JZ ADI STAX JMP : TEST F MGV CPI JZ CPI JZ	H OAF NXRO OOI D PCUR FOR FUNCTION A. E OOI START OOD CR	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR W . STORE CHAP W TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN
71 72 0076 B340 73 0078 B80 74 75 76 77 007A 11603F 72 0070 210030 79 0080 010090 80 81 82 83 0083 FB BACK: 84 0084 C38300 85 86	OUT 040 IN 080 ;SET UP FOINT LXI D.03F LXI H.03S LXI B.090 ;WAIT LOOP FO EI JMP BACK ,HOME UP CURS	; N R S. INTERRUPT CLEAR ; MEYBOARD INTERRUPT CLEAR TERS F60 ; POINT D-E TO REFERENCE TAI D00 ; POINT H-L TO 1ST RAM LOCA' D00 ; POINT B-C TO ACE OR INTERUPTS , ENABLE INTERRUPTED SOR	© 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CARE01 209 016A C601 210 016C 12 LE 211 016D C3B301 I↑ 212 213 214 215 0170 7B FUNC- 216 0171 FE01 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CAFE02	LDAX INX CPI JZ ADI STAX JMP : TEST F MGV CPI JZ CPI JZ CPI JZ	H O4F NXRO O01 D PCUR FOR FUNCTION A.E O01 START O0D CR O11 SAVRO	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR W . STORE CHAP W TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW W CNTL Q (DCI)
71 72 0076 B340 73 0078 B850 74 75 76 77 007A 11603F 72 007D 210030 79 0080 010090 80 81 82 83 0083 FB BACK: 84 0084 C38300 85 86 87 88 0087 210050 HMCUR 89 008A 3E02	OUT 040 IN 080 SET UP FOINT LXI D.0SP LXI H.033 LXI B.090 WALT LOOP FO EI JMP BACK HOME UP CURS LXI H.050 MVI A.002	IN R S INTERRUPT CLEAR KEYBOARD INTERRUPT CLEAR TERS FOO FOINT DE TO REFERENCE TAI DOO FOINT HELTO IST RAM LOCAT OF INTERRUPTS FENABLE INTERRUPTS FOOD UNTIL INTERRUPTED SOR OOO FOINT BEC TO CRTC FOINT BEC TO CRTC TO P REGISTER SELECT	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 1^ 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA0000 218 0176 FE0D 219 0178 CA6E02 220 017B FE11 221 017D CAFE02 222 0130 FE0C 223 0192 CA6101	LDAX INX CPI JZ ADI STAX JMP ; TEST F MGV CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR SAVRO OOC ADCUR	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR W . STORE CHAP W TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SCH) . CARRAGE RETURN . SAVE ROW W CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF)
71 72 0076 0340 73 0078 DES0 74 75 75 75 76 77 007A 11603F 72 007E 210030 79 0080 010090 80 162 83 0083 FB BACK: 84 0084 C38300 85 86 87 83 0097 210050 HMCUR	OUT 040 IN 080 SET UP FOINT LXI D.03F LXI H.03S LXI E.090 ;WAIT LOOP FO EI JMP BACK , HOME UP CURS LXI H.050	IN R S INTERRUPT CLEAR IKEYBOARD INTERRUPT CLEAR TERS F60	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CAREO1 209 016A C601 210 016C 12 LE 211 016D C38301 1^ 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA0000 218 0176 FE0D 219 0178 C46E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0152 CA6101 224 0185 FE0C 225 0187 CAA442	LDAX INX CPI JZ ADI STAX JMP : TEST F HGV CPI JZ CPI JZ CPI JZ CPI	H O4F NXRO O01 D PCUR FOR FUNCTION A.E O01 START O0B CR O11 SAVRO O0C ADCUR O02 HOME	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP & TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW & CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX)
71 72 0076 0340 73 0078 DES0 74 75 76 77 007A 11603F 78 007K 210030 79 0080 010090 80 81 82 83 0083 FE BACK: 84 0084 C38300 B5 86 87 88 0087 210050 HHCUR 89 098A 3E02 90 008C 77 91 008B 3C 92 008E 77	OUT 040 IN 080 ; SET UP FOINT LX1 D. 036 LX1 H. 030 LX1 E. 090 ; WAIT LOOP FO E1 JMP BACK . HOME UP CURS LX1 H. 050 MVI A. 002 MOV M. A. INR A INR A	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR FEO POINT D-E TO REFERENCE TAI DOO POINT H-L TO 1ST RAH LOCA' OR INTERUPTS PENABLE INTERRUPTS LOOP UNTIL INTERRUPTED SOR OOO (""" POINT B-C TO CRTC TO P. REGISTER SELECT TO P. LOAD CURSOR REGISTER SELECT (CURSOR REGISTER SELECT (CURSOR LOADS TO T O P.	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0178 CA6E02 220 0178 FE11 221 0170 CAFE02 220 0178 FE11 221 0170 CAFE02 222 0130 FE0C 223 0132 CA6101 224 0185 FE0C 225 0187 CAB402 226 0184 FE1A	LDAX INX CPI JZ ADI STAX JMP : TEST F HGV CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ	H OAF OAF OAF OAF OAF OAF OO1 D PCUR FOR FUNCTION A. E OO1 START OOD CR OO1 SAVRO OOC ADCUR OO2	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR W . STORE CHAP W TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SCH) . CARRAGE RETURN . SAVE ROW W CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF)
71 72 0076 0340 73 0078 0880 74 75 76 77 77 78 77 78 78 79 79 79 79 79 79 79 79 79 79 79 79 79	OUT 040 IN 080 ; SET UP FOINT LX1 D. 036 LX1 H. 030 LX1 E. 090 ; WAIT LOOP FO E1 JMP BACK . HOME UP CURS LX1 H. 050 MVI A. 002 MOV M. A. INR A INR A	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS F60 : POINT D-E TO REFERENCE TAI 000 : FOINT H-L TO 1ST RAH LOCA' 000 : FOINT B-C TO ACE OR INTERUPTS . ENABLE INTERRUPTS . LOOP UNTIL INTERRUPTED SOR 000 (9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC 216 0171 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0132 CA6101 224 0185 FE00 225 0187 CAA402 226 0187 FE1A 227 0190 CAB502 228 0195 FE0A	LDAX INX INX CPI JZ ADI STAX JMP ; TEST F HOV CPI JZ CPI	H OAF NXRO OOI D PCUR FOR FUNCTION A. E OOI START OOD CR OII SAVRO OOC ADCUR OO2 HOME OIA	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP & TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW & CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX)
71 72 0076 0340 73 0078 DES0 74 75 76 77 77 77 78 77 78 79 79 79 79 79 79 79 79 79 79 79 79 79	OUT 040 IN 080 ; SET UP FOINT LXI D.056 LXI H.050 ; WAIT LOOP FO EI JMP BACK .HOME UP CURS LXI H.050 MVI A.002 MOV H.A 1NR A H0V H.A LXI H.050	IN R S. INTERRUPT CLEAR KEYBOARD INTERRUPT CLEAR FEO POINT D-E TO REFERENCE TAI DOO POINT H-L TO 1ST RAM LOCAT DOO POINT B-C TO ACE OR INTERUPTS ENABLE INTERRUPTED SOR OOO (9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA0000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CA7B02 222 0130 FE0C 223 0152 CA6101 224 0155 FE0C 225 0187 CAB402 226 0187 FEAC 227 0190 CAB502 228 0187 FE0A 229 0191 CASD02 230 0194 FE0S	LDAX INX CPI JZ ADI STAX JMP : TEST F HOV CPI JZ C CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ C CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI DZ C D D D D D D C D D D D D D D D D D	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0D CR O11 SAVRO O0C ADCUR O02 HOME O1A SWAP O0A LF	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL Q (DC1) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUE)
71 72 7076 73 7078 75 75 76 77 77 77 78 78 79 79 79 79 79 79 79 79 79 79 79 79 79	OUT	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS FOO :FOINT D-E TO REFERENCE TAI DOO :FOINT H-L TO IST RAM LOCA' OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR OOO (-^2 :FOINT B-C TO CRTC 2 :T. O P. REGISTER SELECT .T. O P. REGISTER SELECT .T. O P. LOAD .CURSOR ROAD .CURSOR ROAD .CURSOR LOADS TO T O P .POINT H-L TO IST RAM ADD .RETURN ELECT .SAVE D-E REGISTERS	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016C 12 212 213 214 215 0170 7B FUNC- 216 0171 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CA7E02 222 0130 FE0C 223 0132 CA6101 224 0125 FE02 225 0187 CA9402 226 0187 FE0C 227 0190 CAE502 228 0187 CA9402 229 0191 CASD02 230 0194 FE0S 231 0194 FE0S 231 0194 FE0S 231 0196 CAE502 232 0199 FE0E	LDAX INX OPI JADIA STAX JMP : TEST F HOV CPI JZ C CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ C CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI JZ CPI DZ CPI JZ CPI JZ CPI JZ CPI DZ DZ CPI DZ C DZ CPI DZ C DZ C DZ C DZ C DZ C DZ C DZ C DZ	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0D CR O11 SAVRO O0C ADCUR O02 HOME O1A SWAP O0A LF O08 DS	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED
71 72 7076 73 7078 75 76 76 77 77 77 77 78 78 78 79 79 79 79 79 79 79 79 79 79 79 79 79	OUT	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR F60	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA00600 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CA7B02 222 0130 FE0C 223 0130 FE0C 223 0130 FE0C 225 0187 CAA402 226 0184 FE1A 227 019C CAB502 229 0191 CAB502 230 0194 FE0S 231 0196 CAE002 233 0199 FE0C 233 0199 FE0C 233 0199 FE0C 233 0199 FEOC 233 0199 FEOC 233 0199 FEOC 233 0198 CAF102 234 0196 FFIOC 233 0199 FEOCE 233 0198 CAF102	LDAX INX INX JZ ADI JZ ADI STAX JMP ; TEST F HOV CPI JZ CP	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O11 SAVRO O0C ADCUR O02 HOME O1A SHAP O0A LF O08 ES O0B UPCUR O19	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS)
71 71 72 0076 0340 73 0078 DES0 74 75 76 77 007A 11603F 78 007D 210030 79 0080 010090 80 81 82 83 0083 FB BACK: 84 0084 C38300 85 86 87 83 0087 210050 HHCUR 89 098A 3E02 90 008C 77 91 008D 3C 92 008E 77 93 008F 210030 94 0092 C9 95 96 97 98 0093 DS BAUD 99 0094 DB40 100 0096 E60F 101 0086 FE00	OUT 040 IN 080 ; SET UP FOINT LX1 D. 03F LX1 H. 030 ; WAIT LOOP FO E1 JMP BACK . MOHE UP CURS LX1 H. 050 MVI A. 002 MOV H. A INR A LX1 H. 030 RET ; BAUD FATE SE PUSH D IN 040 ANI 046 ANI 046 CPI 000	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR F60	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CAREO1 209 016A C601 210 016C 12 LE 211 016D C3B301 11 212 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA0000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CA7E02 222 0180 FE0C 223 0182 CA6101 224 0185 FE0C 225 0187 CA6402 226 0187 FE11 227 0190 CAE502 228 0187 FE0C 228 0187 FE0C 228 0187 FE0C 229 0191 CA8002 230 0194 FE0S 231 0194 FE0S 231 0196 CAF022 233 0196 CAF102 233 0196 CAF102 234 0196 FE18 235 0196 CA5003	LDAX INX CPI JZ ADI STAX JMP HGV CPI JZ CPI ZZ CPI	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O11 SAURO O0C ADCUR O02 HOME O14B SMAP O00A LF O08 BS O08 UPCUR	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR ; HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) ; UP CURSOR CNTL F (VT)
71 72 0076 0340 73 0078 DES0 74 75 75 76 77 007A 11603F 75 0075 210030 79 0080 010090 80 81 82 83 0083 FB BACK: 84 0084 C38300 85 86 87 88 0087 210050 HHCUR 89 008A 2500 2006C 77 91 008D 3C 92 006E 77 93 008F 210030 90 0092 C9 90 0	OUT 040 IN 080 ; SET UP FOINT LXI D. 03F LXI H. 030 LXI B. 090 ; WAIT LOOP FO EI JMP BACK . HOHE UP CURS LXI H. 050 MVI A. 002 MOV H. A LXI H. 030 RET : BAUD RATE SE PUSH D IN 040 ANI 00F CPI 000 JZ B110 CPI 001	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR FEO : POINT D-E TO REFERENCE TAI DOO : FOINT H-L TO 1ST RAH LOCA' DOR INTERUPTS . ENABLE INTERRUPTS . LOOP UNTIL INTERRUPTED SOR OOO ("" . POINT B-C TO CRTC 2	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 11 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0178 CA6E02 220 0178 FE11 221 0170 CA7E02 222 0180 FE0C 223 0182 CA6101 224 0185 FE0C 225 0187 CA6402 226 0184 FE1A 227 0190 CAE502 228 0191 CA8502 229 0191 CA8502 230 0194 FE08 231 0194 CAE002 233 0194 FE08 231 0194 FE08 231 0198 CAF102 234 0198 FE1B 235 0199 FE6F 233 0198 CAF102 234 0198 FE1B 235 0190 CAF002 234 0198 FE1B 235 0190 CAF002 234 0198 FE1B 235 0190 CAF002 234 0198 FE1B 235 0190 CAF003 236 0143 FE07 237 0145 CA4503	LDAX INX CPI JZ ADI STAX JMP HGV CPI JZ	H O4F NXRO O4F NXRO O61 D PCUR FOR FUNCTION A. E O61 START O60 CR O11 SANRO O60 ADCUR ADCUR O60 ADCUR ADCUR O60 ADCUR O60 ADCUR ADCUR O60 ADCUR A	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL F (VT) . CLEAR ROW CNTL X (CAN) . RINS BELL CNTL G (BEL)
71 72 7076 73 0078 DES0 73 0078 DES0 74 75 76 77 007A 11603F 78 007B 210030 90 80 81 82 83 0083 FE 84 0084 C38300 85 86 87 88 0087 210050 HHCUR 89 0086 77 91 008D 3C 92 008C 77 93 008F 210050 HHCUR 99 009C 77 91 008D 3C 92 008C 77 93 009F 210050 HHCUR 90 90 90 90 90 90 90 90 90 90 90 90 90	OUT 040 IN 080 ; SET UP FOINT LXI D.03F LXI H.03C ; WAIT LOOP FO EI JMP BACK . HOHE UP CURS LXI H.05C MVI A.002 MOV M.A LXI H.05C MOV M.A LXI H.03C RET ; BAUD RATE SE PUSH D IN 040 ANI 00F CPI 000 JZ EII0 OCPI 001 JZ EI5C CPI 002	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR FEO : POINT D-E TO REFERENCE TAI DOO : POINT H-L TO 1ST RAH LOCA' OR INTERUPTS . ENABLE INTERRUPTS . LOOP UNTIL INTERRUPTED SOR . POINT B-C TO CRTC 2	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B 216 0171 FE01 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0132 CA6101 224 0185 FE0C 225 0187 CA0402 226 0187 FE0A 227 0190 CAB502 228 0191 CAB502 229 0191 CAB502 230 0194 FE0S 231 0196 CAE002 232 0198 FE0E 233 0198 CAF102 234 0198 FE0E 234 0196 FE0E 233 0198 CAF102 234 0196 FE0E 234 0196 FE0E 235 0190 CAF003 236 0143 FE0E 237 0190 CAF003 236 0143 FE0E 237 0190 FE0E 233 0198 FE0E 233 0198 FE0E 234 0196 FE0E 235 0140 CAF003 236 0143 FE0T 237 0145 CA4503 238 0148 FE12 239 0144 CA4603	LDAX INX CPI JZ ADI STAX JMP HGV CPI JZ CPI	H O4F NXRO O4F NXRO O61 D PCUR FOR FUNCTION A. E O61 START O60 CR O11 SAURO O60 ADCUR	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL O (DC1) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAF CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL F (VT) . CLEAR ROW CNTL X (CAN) . RING BELL (NTL G (BEL) . INVERT NEXT CNTL R (DC2)
71 72 7076 73 7078 75 76 76 77 77 77 77 77 77 77 77 77 77 77	OUT 040 IN 080 SET UP FOINT LXI D.0SF LXI H.0SSF LXI B.690 WALT LOOP FO EI JMP BACH HOME UP CURS LXI H.0SS MVI A.090 MVI	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR FEO : POINT D-E TO REFERENCE TAI DOO : POINT H-L TO 1ST RAH LOCA' DOO : POINT B-C TO ACE OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR DOO (""" .POINT B-C TO CRIC 2	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 211 016D C3B301 212 213 214 215 0170 7B 216 0171 FE01 217 0178 CA00000 218 0178 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7B02 222 0130 FE0C 223 0132 CA6101 224 0178 FE0C 225 0187 CA6402 226 0187 FE0C 227 0190 CABE02 228 0182 CA6101 224 0185 FE0C 228 0182 CA6101 224 0185 FE0C 228 0182 CA6101 224 0185 FE0C 233 0194 FE0S 231 0196 CAE002 233 0194 FE0S 231 0196 CAE002 234 0196 FE0C 233 0196 CAE002 234 0196 FE0C 234 0197 FE0C 235 0140 CAE003 236 0143 FE0T 237 0145 CA4503 238 0148 FE12 239 0148 CA4503 234 0148 FE13 239 0148 CA4503 240 0148 FE13 241 0148 CA5403	LDAX INX CPI JZ ADI STAX JMP : TEST F HOV CPI JZ CP	H O4F NXRO O01 D PCUR FOR FUNCTION A.E O01 START O0B CR O11 SAVRO O0C ADCUR O02 HOME O1A SMAP O0A LF O08 BS O0B UPCUR O12 CLROW O07 BELL O12	NEXT CHAR LOCATION LAST CHAR OF ROW? LAST CHAR OF ROW? LIFT TRUE JUMP TO NEXT ROW INCREMENT CHAR W STORE CHAP W TO RAM REF. PUT CURSOR HOME AND CLEAR CNTL A (SOM) CARRAGE RETURN SAVE ROW W CNTL Q (DCI) ADVANCE CURSOR CNTL L (FF) HOME UP CNTL B (STX) SHAP CNTL Z (SUB) LINEFEED BACKSPACE CNTL H (RS) JUP CURSOR CNTL F (VT) CLEAR ROW CNTL X (CAN) RING BELL CNTL G (BEL) INVERT NEXT CNTL R (DC2)
71 72 7076 73 7078 75 76 76 77 7078 71 7078 71 7078 71 7078 71 7078 71 7078 71 7078 71 70078 71 70078 71 70078 71 70078 71 70078 71 70078 71 70078 71 70078 71 70090 80 81 82 83 8083 85 84 87 88 8097 81 8097 88 8097 88 8097 88 8097 88 8097 88 8098 87 91 9088 87 91 9088 87 91 9088 87 91 9088 97 98 9093 95 96 97 98 9093 95 96 97 98 9093 95 96 97 98 9093 95 96 97 98 9093 95 96 97 98 9093 95 96 97 98 9093 95 96 97 98 9093 98 9094 98 9096 97 98 9097 98 9098 9099 9099 9099 9099	OUT 040 IN 080 ; SET UP FOINT LX1 D.036 LX1 H.030 LX1 E.090 ; WAIT LOOP FO EI JMP BACH: , HOME UP CURS LX1 H.050 MVI A.002 MOV M. A. INR A MOV M. A. INR CA ANI 006 CPI 000 JN 040 ANI 006 CPI 000 JZ B110 CPI 001 JZ B150 CPI 002 JZ B300	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR FEO : POINT D-E TO REFERENCE TAI DOO : POINT H-L TO 1ST RAH LOCA' OR INTERUPTS . ENABLE INTERRUPTS . LOOP UNTIL INTERRUPTED SOR . POINT B-C TO CRTC 2	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 1 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA0060 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0136 FE0C 223 0136 FE0C 224 0185 FE0C 225 0187 CAA402 226 0187 FEBA 227 0190 CAB502 228 018F FEGA 227 0190 CAB502 228 018F FEGA 227 0190 CAB502 228 018F FEGA 227 0190 CAB502 230 0194 FE0S 231 0194 CAE002 233 0198 CAF102 234 0195 FEOC 233 0198 CAF103 237 0140 CAB002 238 0187 FEOS 231 0194 CAE002 237 0197 FEOS 231 0198 CAF102 238 0198 FEOC 237 0199 FEOC 233 0198 FEOC 233 0198 CAF102 234 0198 FEOS 234 0198 FEOS 235 0140 CA9003 236 0143 FEOT 237 0148 CA4503 238 0148 FE12 239 0140 CA4603 240 0140 FE13	LDAX INX CPI JZ ADI STAX JMP : TEST F HOV CPI JZ CP	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O11 SAVRO O0C ADCUR O02 H OME O1A SHAP O0A LF O08 BS O0B UPCUR O12 CLROH CO7 BELL O12 IVERTN O13 IVERTR	NEXT CHAR LOCATION LAST CHAR OF ROW? LAST CHAR OF ROW? LIFT TRUE JUMP TO NEXT ROW INCREMENT CHAR W STORE CHAP W TO RAM REF. PUT CURSOR HOME AND CLEAR CNTL A (SOM) CARRAGE RETURN SAVE ROW W CNTL Q (DCI) ADVANCE CURSOR CNTL L (FF) HOME UP CNTL B (STX) SHAP CNTL Z (SUB) LINEFEED BACKSPACE CNTL H (PS) JUP CURSOR CNTL F (VT) CLEAR ROW CNTL X (CAN) RING BELL CNTL G (BEL) INVERT NEXT CNTL R (DC2) INVERT ROW CNTL S (DCS) RETURN
71 72 7076 73 7078 75 76 76 77 7078 78 78 77 7078 71 70078 71 70078 71 70078 71 70078 71 70078 71 70078 71 70090 71 70090 71 70090 71 70090 71 70090 71 70090 71 70090 71 70090 71 71 71 71 71 71 71 71 71 71 71 71 71	OUT 040 IN 080 ; SET UP FOINT LX1 D. 05F LX1 H. 050 LX1 E. 090 ; WAIT LOOP FO E1 JMP BACK . HOME UP CURS LX1 H. 050 MVI A. 002 MVI A. 002 MVI A. 002 MVI A. 002 LX1 H. 030 RET : BAUD FATE SE PUSH D IN 040 ANI 00F CPI 000 JZ E110 CPI 001 JZ E150 CPI 002 JZ E300 CPI 003 JZ E500 CPI 003 JZ E500 CPI 003 JZ E500 CPI 004 JZ E500 CPI 005 JZ E500 CPI 004 JZ E500 CPI 004 JZ E500 CPI 005 JZ E500 CPI 004 JZ E500 CPI 004 JZ E500 CPI 005 JZ E500 CPI 004 JZ E500 CPI 005 JZ E500 CPI 004 JZ E500 CPI 005 C	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS FOO :FOINT D-E TO REFERENCE TAI DOO :FOINT H-L TO IST RAM LOCA' OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR .POINT B-C TO CRTC .T. O P. REGISTER SELECT .T. O P. REGISTER SELECT .T. O P. REGISTER SELECT .T. O P. LOAD .CUMSOR ROAD TO TO PPOINT H-L TO IST RAM ADD .RETURN ELECT .SAVE D-E REGISTERS .READ BAUD SELECT CODE .ZERO THE HIGH ORDER 4 BITS .HID BAUD ROUTINE .300 EAUD ROUTINE .300 EAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 11 212 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CAD0000 218 0176 FEDD 219 0178 CAECO2 220 0178 FE11 221 017D CAFE02 222 0130 FE0C 223 0132 CAE101 224 0185 FE0C 225 0187 CAB402 226 0187 CAB402 226 0187 CAB402 226 0187 FE1B 227 0190 CAB502 228 0187 FE0B 229 0191 CASD02 230 0194 FE0S 231 0196 CAECO2 233 0198 FE1D 234 0198 FE1B 235 0189 CAFT02 234 0198 FE1B 235 0189 CAFT02 234 0198 FE1B 235 0180 CAF502 238 0184 FE1C 239 0185 CAFT02 239 0194 CAFF03 233 0186 FE1C 239 0186 CAFF02 234 0196 FE1B 235 0187 CAFF03 233 0188 FE1C 239 0184 CAFF03 233 0188 FE1C 239 0184 CAFF03 233 0184 FE1C 239 0184 CAFF03 234 0185 CAFF03 234 0185 CAFF03 234 0186 CAFF03 235 0186 CAFF03 237 0187 CAFF03 23	LDAX INX CPI JZ ADI STAX JMP : TEST F HOV CPI JZ CP	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O11 SAVRO O0C ADCUR O02 H OME O1A SHAP O0A LF O08 BS O0B UPCUR O12 CLROH CO7 BELL O12 IVERTN O13 IVERTR	NEXT CHAR LOCATION LAST CHAR OF ROW? LAST CHAR OF ROW? LIFT TRUE JUMP TO NEXT ROW INCREMENT CHAR W STORE CHAP W TO RAM REF. PUT CURSOR HOME AND CLEAR CNTL A (SOM) CARRAGE RETURN SAVE ROW W CNTL Q (DCI) ADVANCE CURSOR CNTL L (FF) HOME UP CNTL B (STX) SHAP CNTL Z (SUB) LINEFEED BACKSPACE CNTL H (RS) JUP CURSOR CNTL F (VT) CLEAR ROW CNTL X (CAN) RING BELL CNTL G (BEL) INVERT NEXT CNTL R (DC2)
71 72 7076 73 7078 75 76 76 77 77 77 77 77 77 77 77 78 77 78 78 78	OUT 040 IN 080 ; SET UP FOINT LX1	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS FEO : FOINT D-E TO REFERENCE TAI DOO: FOINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR .FOINT B-C TO CRTC .T. O. P. REGISTER SELECT .T. O. P. LOAD .CURSOR REGISTER SELECT .CURSOR REGISTER SELECT .CURSOR LOADS TO T O. PPOINT H-L TO 1ST RAH ADD .RETURN ELECT .SAVE D-E REGISTERS .READ BAUD SELECT CODE .ZERO THE HIGH ORDER 4 BITS .110 BAUD ROUTINE .300 BAUD ROUTINE .400 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 211 016D C3B301 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA0000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CAFE02 222 0180 FE0C 223 0182 CA6101 224 0185 FE0C 225 0187 CAA402 226 0184 FE1A 227 0190 CAE502 228 0187 FE0B 229 0191 CAB002 230 0194 FE0B 231 0194 FE0B 231 0194 FE0B 231 0194 FE0B 233 0198 FE1C 233 0198 FE1C 233 0198 FE1C 233 0198 FE1C 234 0198 FE1B 235 0140 CA5003 236 0143 FE07 237 0145 CA4503 238 0148 FE1C 239 0140 CA5003 234 0140 FE13 241 0146 CA5403 242 0182 C9 244 245	LDAX INX CPI JZ ADI STAX JMP HGV CPI JZ CPI	H O4F NXRO O4F NXRO O501 D PCUR FOR FUNCTION A. E O51 START O50 CR O11 SAURO O50 ADCUR O50 ADCU	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (VT) . CLEAR ROW CNTL X (CAN) . RING BELL CNTL G (BEL) . INVERT NEXT CNTL R (DC2) . INVERT ROW CNTL S (DC3) . RETURN . REG TO ACC
71 72 7076 73 7078 75 76 76 77 77 77 77 77 77 77 77 77 77 77	OUT 040 IN 080 ; SET UP FOINT LX1 D.03F LX1 H.03C LX1 E.09C ; WAIT LOOP FO EI JMP BACK . HOHE UP CURS LX1 H.05C MVI A.002 MOV H.A INR A LX1 H.03C RET ; BAUD RATE SE FUSH D IN 040 ANI 00F CPI 000 JZ E110 CPI 001 JZ E150 CPI 002 JZ E300 CPI 003 JZ E300 CPI 004 JZ E300 CPI 004 JZ E300 CPI 005 JZ E300 CPI 004 JZ E300 CPI 005 JZ E300 CPI 005 JZ E1200 CPI 005 JZ E1200 CPI 005 JZ E1200 CPI 006	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS) FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS ENABLE INTERRUPTS ENABLE INTERRUPTS FOINT B-C TO CRTC TO P. REGISTER SELECT TO P. LOAD CURSOR REGISTER SELECT CURSOR REGISTER SELECT CURSOR REGISTER SELECT CURSOR LOADS TO TO P POINT H-L TO 1ST RAM ADD RETURN ELECT SAVE D-E REGISTERS READ BAUD SELECT CODE TERO THE HIGH ORDER A BITS TO BAUD ROUTINE SOD BAUD ROUTINE 1200 BAUD ROUTINE 1200 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 211 016D C3B301 212 213 214 215 0170 7B 216 0171 FE01 217 0178 CA6602 218 0176 FE0D 219 0178 CA6602 220 0178 FE11 221 0170 CAFE02 222 0180 FE0C 223 0182 CA6101 224 0185 FE0C 225 0187 CAA402 226 0187 CAA402 226 0187 CAA402 227 0190 CAE502 228 0187 FE1A 227 0190 CAE502 228 0187 FE0E 229 0191 CASD02 230 0194 FE0S 231 0194 CAE502 233 0198 CAFT02 233 0198 CAFT02 233 0198 CAFT02 234 0196 FE18 235 0140 CA5003 236 0143 FE07 237 0145 CA4503 238 0148 FE12 239 0140 CA5003 236 0143 FE07 237 0145 CA4503 238 0148 FE12 239 0140 CA5003 234 0148 FE12 239 0140 CA5003 234 0146 CA5403 234 0147 CA5403 234 0148 CA5403 244 0148 CA5403 245 0183 7C 247 0184 C620 248 0183 7C 247 0184 C620 248 0183 7C 247 0184 C620 248 0185 7C 247 0184 C620 248 0185 7C 247 0184 C620 248 0185 7C 247 0184 C620 248 0186 67	LDAX INX CPI JZ ADI STAX JMP HOV CPI JZ CPI	H O4F NXRO O4F NXRO O501 D PCUR FOR FUNCTION A. E O51 START O50 CR O11 SAURO O50 ADCUR O50 ADCU	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (VT) . CLEAR ROW CNTL X (CAN) . RINS BELL CNTL G (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . ROM H-L REGISTERS . H REG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD.
71 72 7076 73 72 73 75 75 76 77 77 75 76 77 77 77 77 78 77 78 78 78 78 78 78 78	OUT 040 IN 080 ;SET UP FOINT LXI D.05F LXI H.030 ;MAIT LOOP FO EI JMP BACK ;MOHE UP CURS LXI H.050 MVI A.002 MVI A.002 MVI H.A INR A LXI H.050 RET ;BAUD RATE SE PUSH D IN 040 ANI 09F CPI 000 JZ EII0 CPI 001 JZ EI20 CPI 002 JZ E300 CPI 003 JZ E300 CPI 004 JZ E120 CPI 005	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS) FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS FOOD UNTIL INTERRUPTED FOOD COMPANY POINT B-C TO CRTC TO P. REGISTER SELECT TO P. LOAD CURSOR REGISTER SELECT CURSOR LOADS TO TO P POINT H-L TO 1ST RAM ADD RETURN ELECT SAVE D-E REGISTERS READ BAUD SELECT CODE TERO THE HIGH ORDER A BITS TO BAUD ROUTINE TO BAUD ROUTINE SOO BAUD ROUTINE 1200 BAUD ROUTINE 1200 BAUD ROUTINE 1200 BAUD ROUTINE 1200 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7B02 222 0130 FE0C 223 0130 FE0C 233 0140 CAB502 234 0185 FE0C 235 0187 CAB502 236 0187 FE0B 231 0196 CAB502 233 0198 CAF002 233 0198 CAF002 234 0196 CAF002 237 0197 FE0E 237 0198 CAF002 234 0198 FE1B 235 01A0 CAF003 236 01A3 FE07 237 01A5 CA4503 236 01A3 FE07 237 01A5 CA4503 236 01A3 FE07 237 01A6 CA4503 240 01AD FE13 241 01AF CA5403 244 01B2 C9 243 244 245 01B3 7C 247 01E4 C620	LDAX INX CPI JZ ADI STAX JMP :TEST F HOV CPI JZ CPI	H O4F NXRO O01 D PCUR FOR FUNCTION A.E O01 START O0B CR O11 SAVRO O0C ADCUR O02 HOME O1A SMAP O0A LF O08 DS DS UPCUR O12 LF O18 CREUN O07 DELL O17 LF O18 CLRUM O07 DELL O17 LF O08 CLRUM O07 DELL	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SWAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (CAN) . RINS BELL CNTL O (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . REG TO ACC . SET H-L REG TO CRTC ADD . H IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM
71 72 7076 73 72 75 75 76 77 75 77 77 77 77 78 77 78 77 78 78 78 78 78	OUT 040 IN 080 I	IN R S INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS FOUNT E-C TO CRTC TO P. REGISTER SELECT TO P. LOAD CURSOR REGISTER SELECT CURSOR REGISTER SELECT CURSOR LOADS TO TO P POINT H-L TO 1ST RAM ADD RETURN ELECT SAVE D-E REGISTER READ BAUD SELECT CODE TERO THE HIGH ORDER 4 BITS TO BAUD ROUTINE TO BAUD ROUTINE TO BAUD ROUTINE 100 BAUD ROUTINE 1200 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 17 212 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA00000 218 0176 FE0D 219 0178 CAEC02 220 0178 FE11 221 017D CATRO2 222 0130 FE0C 223 0130 FE0C 223 0130 FE0C 224 0185 FE0C 225 0187 CAA402 226 0187 FE1A 227 0196 CAB502 238 0197 CAB502 230 0194 FE0B 231 0196 CAEC02 233 0198 FE0C 233 0198 CAF102 234 0198 FE1B 235 01A0 CAEC02 234 0198 FE1B 235 01A0 CAEC02 234 0198 FE0C 233 0198 CAF102 234 0198 CAF003 245 0180 TC 247 0184 CAEC0 248 0188 TC 247 0184 CAEC0 248 0188 TC 247 0184 CAEC0 255 0189 TC	LDAX INX CPI JZ ADIA STAX JMP :TEST F HOV CPI JZ CP	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O01 SAVRO O0C ADCUR O02 HOME O1A SWAPO O0A LF O08 DS DS O0R UPCUR O12 LF O08 DS UPCUR O12 IVERTN O13 IVERTR CURSOR TO CRIC F A. H O20 H. A H. O03 A. H O02	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP & TO RAH REF PUT CURSOR . HOME AND CLEAR CNTL A (SOH) . CARRAGE RETURN . SAVE ROW & CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL F (VT) . CLEAR ROW CNTL X (CAN) . RINS BELL CNTL G (BEL) . INVERT NEXT CNTL R (DC2) . INVERT ROW (NTL S (DC3) . RETURN . H REG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD CURSOR RECISTER SELECT
71 72 7076 73 7078 75 75 76 77 77 77 77 78 78 78 79 79 79 79 79 79 79 79 79 79 79 79 79	OUT 040 IN 080 (SET UP FOINT LXI D. 095 LXI H. 030 LXI B. 090 (MAIT LOOP FO EI JMP BACK: (MOHE UP CURS LXI H. 030 MVI A. 002 MOV H. A. 003	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS) FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS FOUNT E-C TO CRTC TO P. REGISTER SELECT TO P. LOAD CURSOR REGISTER SELECT CURSOR REGISTER SELECT CURSOR LOADS TO TO P POINT H-L TO 1ST RAM ADD RETURN ELECT SAVE D-E REGISTERS READ BAUD SELECT CODE TERO THE HIGH ORDER 4 BITS TO BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 1 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0132 FE0C 223 0132 FE0C 224 0185 FE0C 225 0187 CAA402 226 018A FE1A 227 019C CAB502 238 019F FE0A 229 0191 CABD02 238 019F FE0A 229 0191 CABD02 238 019F FE0A 239 0194 FE0B 231 0196 CAE002 233 0198 CAF102 234 0198 CAF102 234 0198 CAF102 234 0198 CAF102 237 0145 CA4503 236 0148 FE12 239 014A CA4603 240 0181 CA4503 240 0183 TC 247 0184 C620 248 0183 TC 247 0184 C620 248 0183 TC 247 0184 C620 255 0186 67	LDAX INX INX JPI ADI JZ ADI STAX JMP : TEST F HOV CPI JZ C	H O4F NXRO O4F NXRO O50 D PCUR FOR FUNCTION A. E O51 START O50 CR O11 SAURO O50 ADCUR O50 ADCUR O51 ADCUR O51 ADCUR O51 ADCUR O52 HOME O51 ADCUR O52 HOME O51 ADCUR O52 HOME O51 ADCUR O53 ADCUR O53 ADCUR O54 ADCUR O55 ADCUR O55 ADCUR O56 ADCUR O57 BELL O57 BELL O58 BELL O5	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SWAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (CAN) . RINS BELL CNTL O (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . REG TO ACC . SET H-L REG TO CRTC ADD . H IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM
71 72 7076 73 7078 75 75 76 77 75 77 77 77 77 78 77 78 78 78 79 79 79 79 79 79 79 79 79 79 79 79 79	OUT 040 IN 080 SET UP FOINT LXI	IN R S INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS ENABLE INTERRUPTS ENABLE INTERRUPTS FOOD FOOD WITH INTERRUPTED FOOD COMPOSE REGISTER SELECT CURSON REGISTERS READ BAUD SELECT CODE ZERO THE HIGH ORDER 4 BITS 110 BAUD ROUTINE 150 BAUD ROUTINE 1600 BAUD ROUTINE 1200 BAUD ROUTINE 1200 BAUD ROUTINE 1300 BAUD ROUTINE 1400 BAUD ROUTINE 1400 BAUD ROUTINE 1500 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 1 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0132 FE0C 223 0132 FE0C 224 0185 FE0C 225 0187 CAA402 226 0187 FE0A 227 019C CAB502 228 018F FEGA 229 0191 CABD02 233 0194 FE0S 231 0196 CABC02 233 0198 CAF102 234 0195 FE0C 233 0198 CAF102 234 0195 FE0C 233 0196 CABC02 237 0197 FE0C 233 0198 CAF102 234 0198 FE18 235 0140 CABC03 236 0143 FE07 237 0145 CA4503 240 0181 FE13 241 0146 CA4503 240 0181 CA4503 240 0181 FE13 241 0146 CA4503 242 0182 C9 243 0186 CAF102 243 0148 FE12 249 0140 CA4603 240 0181 FE13 241 0146 CA5403 242 0182 C9 243 0186 C7 247 0184 C620 248 0183 7C 247 0184 C620 248 0187 7C 251 0186 C7 255 0180 C9 255	LDAX LDAX INX INX ADI JZ ADI STAX JMP STEST F HOV CPI JZ C	H O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F PUR FOR FUNCTION A. E O4F	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (BS) . UP CURSOR CNTL K (CAN) . RINS BELL CNTL G (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . ROM H-L REGISTERS . M REG TO ACC . SET H-L REG TO CRTC ADD . M IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM . ADDRESS
71 72 7076 73 7078 75 76 76 77 77 77 77 77 77 77 77 77 77 77	OUT	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR TERS) FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS ENABLE INTERRUPTS ENABLE INTERRUPTS FOOD WITH INTERRUPTED FOOD WITH INTERRUPTED OO COMPOSE REGISTER SELECT (TO P. LOAD (CURSOR REGISTER SELECT (CURSOR LOADS TO TO P. POINT H-L TO 1ST RAM ADD RETURN ELECT SAVE D-E REGISTERS READ BAUD SELECT CODE (ZERO THE HIGH ORDER 4 BITS) 110 BAUD ROUTINE 150 BAUD ROUTINE 1600 BAUD ROUTINE 1200 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 1 213 214 215 0170 7B FUNC 216 0171 FE0D 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7E02 222 0130 FE0C 223 0132 FE0C 223 0132 FE0C 224 0185 FE0C 225 0187 CAA402 226 018A FE1A 227 019C CAB502 228 018F FEGA 229 0191 CAB002 233 0198 CAF102 233 0198 CAF102 233 0198 CAF102 234 0195 FE0C 233 0198 CAF102 234 0195 FE0C 233 0198 CAF102 234 0196 CAB002 237 0145 CA5003 236 0140 CA5003 236 0143 FE07 237 0145 CA4503 240 0181 FE13 241 0146 CA5003 240 0181 FC7 247 0148 C620 243 018 CAF102 244 0183 TC 247 0184 C620 244 0183 TC 247 0184 C620 248 0185 TC 247 0184 C620 248 0186 TC 247 0186 C620 255 0186 TC	LDAX INX CPI JZ ADI STAX JMP FEST F HGV CPI JZ CP	H O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F PUR FOR FUNCTION A. E O4F	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP # TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW # CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (CAN) . RING BELL CNTL G (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . HEG TO ACC . SET H-L REG TO CRTC ADD . H IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM . ADDRESS . RETURN .
71 72 72 73 73 74 75 76 76 77 77 77 77 77 77 77 77 77 77 77	OUT	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR FEO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR .POINT E-C TO CRTC .T. O. P. REGISTER SELECT .T. O. P. LOAD .CURSOR REGISTER SELECT .CURSOR REGISTER SELECT .CURSOR LOADS TO TO P. POINT H-L TO 1ST RAM ADD .RETURN ELECT .SAVE D-E REGISTERS .READ BAUD SELECT CODE .ZERO THE HIGH ORDER 4 BITS .110 BAUD ROUTINE .150 BAUD ROUTINE .200 BAUD ROUTINE .1200 BAUD ROUTINE .1200 BAUD ROUTINE .2400 BAUD ROUTINE .2400 BAUD ROUTINE .3400 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 211 016D C3B301 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CA7B02 222 0130 FE0C 223 0187 CA6101 224 0178 CA6E02 225 0187 CA6101 224 0185 FE0C 225 0187 CA6402 226 0187 CA6402 227 0190 CAB502 228 0187 FE0C 228 0187 FE0C 228 0187 FE0C 233 0194 FE0S 231 0194 CAE002 230 0194 FE0S 231 0194 CAE002 230 0195 CAF002 230 0196 CAF002 230 0197 FE0C 233 0198 CAF102 234 0196 FE13 235 0140 CA5003 236 0148 FE12 239 0149 FE13 241 0148 CA503 243 244 245 246 0183 7C 247 0184 C620 243 0186 67 249 0186 67 255 0186 CDDC01 NXR0	LDAX INX CPI JZ ADI STAX JMP :TEST F HGV CPI JZ CPI	H O4F NXRO O4F NXRO O50 D PCUR FOR FUNCTION A. E O51 START O50 CR O11 SAURO O50 CR O11 SAURO O50 CR	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (BS) . UP CURSOR CNTL K (CAN) . RINS BELL CNTL G (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . ROM H-L REGISTERS . M REG TO ACC . SET H-L REG TO CRTC ADD . M IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM . ADDRESS
71 72 7076 73 7078 75 76 77 77 77 77 77 77 77 77 77 77 77 77	OUT 040 IN 080 IN 080 IN 080 ISET UP FOINT LXI D. 095 LXI H. 030 LXI B. 090 IN MAIT LOOP FO EI JMP BACK IN A 030 MOV H. A LXI H. 030 MOV H. A LXI	IN R S. INTERRUPT CLEAR (KEYBOARD INTERRUPT CLEAR FEO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR .POINT E-C TO CRTC .T. O. P. REGISTER SELECT .T. O. P. LOAD .CURSOR REGISTER SELECT .CURSOR REGISTER SELECT .CURSOR LOADS TO TO P. POINT H-L TO 1ST RAM ADD .RETURN ELECT .SAVE D-E REGISTERS .READ BAUD SELECT CODE .ZERO THE HIGH ORDER 4 BITS .110 BAUD ROUTINE .150 BAUD ROUTINE .200 BAUD ROUTINE .1200 BAUD ROUTINE .1200 BAUD ROUTINE .2400 BAUD ROUTINE .2400 BAUD ROUTINE .3400 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 LT 212 LE 211 016D C3B301 LT 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA00000 218 0176 FE0B 219 0178 CA6E02 220 0178 FE11 221 017D CA7B02 222 0130 FE0C 223 0135 CA6401 224 0185 FE02 225 0187 CAA402 226 0187 CAA402 227 0190 CAB502 230 0194 FE08 231 0196 CAE502 233 0198 CAF102 233 0198 CAF102 234 0196 CAE502 234 0196 CAE502 237 0196 CAE502 237 0197 CAB502 230 0194 FE08 231 0196 CAE502 232 0197 FE0E 233 0198 CAF102 234 0196 CAF002 234 0196 FE0 235 0196 CAF002 234 0196 FE1 235 01A0 CA5003 236 01A3 FE07 237 01A3 CA4503 238 01A3 FE07 237 01A3 CA4503 238 01A3 FE07 237 01A5 CA4503 238 01A8 FE12 239 01AA CA4803 240 01AB FE13 241 01AB FE13 241 01AF CA5003 250 01AF	LDAX INX CPI JZ ADI STAX JMP TEST F HOV CPI JZ CPI	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O11 SAVRO O0C ADCUR O02 HOME O1A SMAP O0A LF O08 BS O0B UPCUR O12 IVERTN O13 IVERTR CURSOR TO CRIC F A. H O2O H. A ROH ON SCREEN NXRO1 ZCHAR H	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP & TO RAH REF PUT CURSOR . HOME AND CLEAR CNTL A (SOH) . CARRAGE RETURN . SAVE ROW & CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (VT) . CLEAR ROW CNTL X (CAN) . RING BELL CNTL G (BEL) . INVERT NEXT CNTL R (DC2) . INVERT ROW (NTL S (DC3) . RETURN . HEG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM . ADDRESS . RETURN . GO TO NEXT ROW SUBPOUTINE . ZAVE H-L . SAVE H-L . STORE ROW . STORE SAVE H-L . STORE ROW . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . STORE ROW SUBPOUTINE . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . STORE ROW SUBPOUTINE . STORE ROW SUBPOUTINE . SAVE H-L . STORE ROW SUBPOUTINE . STORE ROW S
71 72 7076 73 7078 75 75 76 77 77 77 77 77 77 77 77 77 77 77 77	OUT	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS) FOO FOINT D-E TO REFERENCE TAI DOO FOINT H-L TO 1ST RAM LOCA' POINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS LENABLE INTERRUPTS LOOP UNTIL INTERRUPTED FOO FOO FOO FOO FOO FOO FOO FOO FOO FO	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 LT 212 LE 211 016D C3B301 LT 212 213 214 215 0170 7B 216 0171 FE01 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 0170 CA7B02 222 0130 FE0C 223 0132 CA6101 224 0185 FE02 225 0187 CAA402 226 0184 FE1A 227 0190 CAB502 230 0194 FE08 231 0196 CAE502 233 0198 CAF002 233 0198 CAF002 233 0198 CAF002 233 0198 CAF002 234 0196 FE0 233 0198 CAF002 237 0196 CA6503 236 0186 FE1 235 01A0 CA6503 236 01A0 FE13 237 01A0 CA6503 237 01A0 CA6503 238 01A0 CA6503 239 01A0 CA6503 239 01A0 CA6503 230 01A0 FE13 241 01AF CA5003 242 01B2 C9 243 244 245 246 01B3 7C 247 01B4 C620 248 01B6 67 249 01B7 3603 250 01BP CD 251 01BP CD 252 01BC CD 253 01BD C9 253 01BD C9 253 01BD C9 253 01BC CD 250 01C1 CDF301 260 01C4 E5 CLR0H3 261 01C5 1E60 262 01C7 1A	LDAX INX CPI JZ ADI STAX JMP TEST F HOV CPI JZ CPI	H O4F NXRO O01 D PCUR FOR FUNCTION A. E O01 START O0B CR O11 SAVRO O0C ADCUR O02 HOME O1A SMAP O0A LF O08 DS O08 UPCUR O12 IVERTN O13 IVERTR CURSOR TO CRIC F A. H O2O H. A ROW ON SCREEN NXROI ZCHARR H E. LASTROW D	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP & TO RAH REF PUT CURSOR . HOME AND CLEAR CNTL A (SOH) . CARRAGE RETURN . SAVE ROW & CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL F (VT) . CLEAR ROW CNTL X (CAN) . RING BELL CNTL G (BEL) . INVERT NEXT CNTL R (DC2) . INVERT ROW (NTL S (DC3) . RETURN . Y . REM H-L REGISTERS . H REG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD H IS CRTC ADD CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM . ADDRESS . RETURN . GO TO NEXT ROW SUBPOLITINE . ZERO CHARACTER . SAVE H-L . FOINT D.E TO LASTROW
71 72 0076 0340 73 0078 0880 74 75 75 76 77 007A 11603F 78 007D 210030 79 0080 010090 80 81 82 84 0084 C38300 85 86 87 88 0087 210050 HHCUR 89 0084 C38300 85 86 87 89 0087 210050 HHCUR 90 008C 77 93 008F 77 98 009 008F 77 98 009 008F 77 98 009 008F 77 98 008F 77 9	OUT 040 IN 080 I	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS FOO : POINT D-E TO REFERENCE TAI DOO: POINT H-L TO 1ST RAM LOCA' POINT H-L TO 1ST RAM LOCA' POINT B-C TO ACE OR INTERUPTS .ENABLE INTERRUPTS .LOOP UNTIL INTERRUPTED SOR .POINT B-C TO CRTC .T. O. P. REGISTER SELECT .T. O. P. LOAD .CURSOR REGISTER SELECT .CURSOR REGISTER .READ BAUD SELECT CODE .ZERO THE HIGH ORDER 4 BITS .110 BAUD ROUTINE .300 BAUD ROUTINE	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 LT 212 LE 211 016D C3B301 LT 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7F02 222 0130 FE0C 223 0132 FE0C 223 0132 FE0C 223 0132 FE0C 223 0136 FE0C 223 0136 FE0C 223 0136 FE0C 223 0137 FE0C 223 0138 FE0C 223 0138 FE0C 223 0138 FE0C 223 0138 FE0C 223 0139 FE0C 233 0198 CAB502 233 0198 CAB502 233 0198 CAF102 234 0196 FE0C 233 0198 CAF102 234 0196 FE0C 234 0196 FE0C 234 0196 FE0C 234 0196 FE0C 234 0197 FE0C 237 0196 CA5003 236 0148 FE12 239 0140 CA5003 236 0148 FE12 239 0140 CA5003 236 0148 FE12 239 0140 CA5003 236 0148 FE0C 237 0148 CA5003 240 0149 FE18 235 0140 CA5003 240 0149 FE18 241 0145 CA5403 242 0150 FF CA5403 244 0150 FF CA5403 245 0150 FF CA5403 246 0150 FF CA5403 247 0150 CA5003 250 0150 C9 253 0150 C9 253 0150 C9 255 0150 C9 255 0150 C9 256 0150 CDDC01 NXRO 257 0150 CBF 256 0150 CDDC01 NXRO 257 0150 CBF 257 0150 CDDC01 NXRO 259 0161 CDF201 259 0161 CDF201 259 0161 CDF201 260 0164 E5 261 0165 1560 262 0167 1A 263 0168 C601 264 0168 FE300	LDAX INX CPI JZ ADI STAX JMP STEST F HOV CPI JZ CPI	H O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F O4F O4F O4F O4F O4F O4F O4F O4F O4	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR & . STORE CHAP & TO RAH REF PUT CURSOR . HOME AND CLEAR CNTL A (SOH) . CARRAGE RETURN . SAVE ROW & CNTL Q (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (VT) . CLEAR ROW CNTL X (CAN) . RING BELL CNTL G (BEL) . INVERT NEXT CNTL R (DC2) . INVERT ROW (NTL S (DC3) . RETURN . HEG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD . CURSOR REGISTER SELECT . H REG SET EAC) TO VIDIO RAM . ADDRESS . RETURN . GO TO NEXT ROW SUBPOUTINE . ZAVE H-L . SAVE H-L . STORE ROW . STORE SAVE H-L . STORE ROW . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . SAVE H-L . STORE ROW SUBPOUTINE . STORE ROW SUBPOUTINE . SAVE H-L . STORE ROW SUBPOUTINE . SAVE H-L . STORE ROW SUBPOUTINE . STORE ROW SUBPOUTINE . SAVE H-L . STORE ROW SUBPOUTINE . STORE ROW S
71 72 7076 73 7078 75 76 76 77 77 77 77 77 77 77 77 77 77 77	OUT	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS F60	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 211 016D C3B301 212 213 214 215 0170 7B 216 0171 FE01 217 0173 CA00000 218 0178 FE0D 219 0178 FE11 221 0170 A7B02 220 0178 FE11 221 0170 A7B02 222 0180 FE0C 223 0182 CA6101 224 0185 FE0C 225 0187 CAA402 226 0187 FEAC 227 0190 CAB502 228 0187 FE0A 229 0191 CAB502 230 0194 FE08 231 0196 CAE502 233 0198 CAF102 234 0196 FE18 235 0140 CAF003 236 0143 FE12 239 0146 CAF003 234 0196 CAF003 235 0190 CAF003 241 0197 A8003 242 0182 C9 243 0187 FE0A 245 0183 7C 247 0184 CAE00 249 0187 3603 240 0140 FE13 241 0146 CAF003 242 0182 C9 243 0185 FC 245 0186 CF 247 0186 CAF003 255 0187 CAF003 255 0180 C9 255 0180 C9 255 0181 CDF001 NXR0 259 01C1 CDF301 260 01C4 E5 261 01C5 1E60 262 01C7 1A 263 01C6 CAF001	LDAX INX CPI JZ ADI STAX HGV CPI JZ	H 04F NXRO 0001 D PCUR FOR FUNCTION A.E 001 START 00D CR 011 SAURO 00C ADCUR	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (CAN) . RINS BELL CNTL G (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . REG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD . CURSOR REGISTER SELECT . H REG TO ACC SET H-L REG TO CRTC ADD H SC CRTC ADD . CURSOR REGISTER SELECT . H REG TO ACC SET H-L REG TO CRTC ADD H SC CRTC ADD . CURSOR REGISTER SELECT . ADDRESS . RETURN . CO TO NEXT ROW SUBPOLITINE . ZERO CHARACTER . SAVE H-L . FOINT D.E TO LASTROW . POINT AC TO FIRST ROW OFF SC
71 72 0076 0340 73 0078 DES0 74 75 75 0078 11603F 75 0076 210030 79 0080 010090 80 81 82 83 0083 FB BACK: 84 0084 C38300 BS 85 86 87 88 0087 210050 HHCUR 89 0984 3802 90 008C 77 91 008D 3C 92 008C 77 93 008F 210030 94 0092 C3 95 96 97 0098 DESC 77 93 008F 210030 94 0097 E011 104 0097 E050 102 0098 E050 102 0098 E050 102 0098 E050 103 0099 FED1 104 0097 E050 105 0082 FED0 115 0085 FED0 115 0085 C4F500 115	OUT	IN R S. INTERRUPT CLEAR (MEYBOARD INTERRUPT CLEAR TERS F60	9 205 0163 1A 206 0164 23 207 0165 FE4F 208 0167 CABE01 209 016A C601 210 016C 12 LE 211 016D C3B301 LT 212 LE 211 016D C3B301 LT 212 213 214 215 0170 7B FUNC 216 0171 FE01 217 0173 CA00000 218 0176 FE0D 219 0178 CA6E02 220 0178 FE11 221 017D CA7F02 222 0130 FE0C 223 0132 FE0C 223 0132 FE0C 223 0132 FE0C 223 0136 FE0C 223 0136 FE0C 223 0136 FE0C 223 0137 FE0C 223 0138 FE0C 223 0138 FE0C 223 0138 FE0C 223 0138 FE0C 223 0139 FE0C 233 0198 CAB502 233 0198 CAB502 233 0198 CAF102 234 0196 FE0C 233 0198 CAF102 234 0196 FE0C 234 0196 FE0C 234 0196 FE0C 234 0196 FE0C 234 0197 FE0C 237 0196 CA5003 236 0148 FE12 239 0140 CA5003 236 0148 FE12 239 0140 CA5003 236 0148 FE12 239 0140 CA5003 236 0148 FE0C 237 0148 CA5003 240 0149 FE18 235 0140 CA5003 240 0149 FE18 241 0145 CA5403 242 0150 FF CA5403 244 0150 FF CA5403 245 0150 FF CA5403 246 0150 FF CA5403 247 0150 CA5003 250 0150 C9 253 0150 C9 253 0150 C9 255 0150 C9 255 0150 C9 256 0150 CDDC01 NXRO 257 0150 CBF 256 0150 CDDC01 NXRO 257 0150 CBF 257 0150 CDDC01 NXRO 259 0161 CDF201 259 0161 CDF201 259 0161 CDF201 260 0164 E5 261 0165 1560 262 0167 1A 263 0168 C601 264 0168 FE300	LDAX INX CPI JZ ADI STAX JMP HOV CPI JZ CPI	H O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F NXRO O4F O4F O4F O4F O4F O4F O4F O4F O4F O4	. NEXT CHAR LOCATION . LAST CHAR OF ROW? . LAST CHAR OF ROW? . IF TRUE JUMP TO NEXT ROW . INCREMENT CHAR * . STORE CHAP * TO RAM REF PUT CURSOR . HOME AND CLEAR CNTL A (SOM) . CARRAGE RETURN . SAVE ROW * CNTL O (DCI) . ADVANCE CURSOR CNTL L (FF) . HOME UP CNTL B (STX) . SHAP CNTL Z (SUB) . LINEFEED . BACKSPACE CNTL H (RS) . UP CURSOR CNTL K (CAN) . RINS BELL CNTL G (BEL) . INVERT NEXT CNTL S (DCS) . RETURN . REG TO ACC SET H-L REG TO CRTC ADD H IS CRTC ADD . CURSOR REGISTER SELECT . H REG TO ACC SET H-L REG TO CRTC ADD H SC CRTC ADD . CURSOR REGISTER SELECT . H REG TO ACC SET H-L REG TO CRTC ADD H SC CRTC ADD . CURSOR REGISTER SELECT . ADDRESS . RETURN . CO TO NEXT ROW SUBPOLITINE . ZERO CHARACTER . SAVE H-L . FOINT D.E TO LASTROW . POINT AC TO FIRST ROW OFF SC

66 OTCF (D8302		CALL	L DHL 1	LOAD H.L WITH ADD OF LASTRO			. CARRA	GE RETURN	
67 01D2 CD3803 68 01D5 E1 69 01D6 C9 70		CALL FOR RET	OUROWS H	, RESTORE H.L	401 402 026E 1E63 403 0270 3E00 404 0272 12	CR	MVI MVI STAX	E. CHARNUM A. 000 D	FOINT D-E TO CHAR #
71 01D7 3E00 72 01D9 C3CF01 73		MVI JMF	A. 000 L00F5	, LOAD ROW ZERO	405 0273 1E61 406 0275 CD9202 407 0278 C3B301 408		MVI CALL JMP	E, ROWSOSO LDHL PCUR	CURSOR TO THE BEGINNING O
74 75		, NEXT I			409		SAVE	ROW	
76 010C 1E60 77 010E 1A 78 010F EB 79 01E0 23 80 01E1 BE	NXR01	HVI LDAX XCHG INX CMP	E, LASTRON D	, POINT D-E REG TO LAST ROW , PUT LAST ROW W TO ACC , EXCHANGE H-L WITH D-E , H-L IS NOW AT 8080 ROW W , COMPARE LAST ROW W WITH , 8080 ROW W IF TRUE SCROLL	411 0278 1E61 412 027D 1A 413 027E 1E65 414 0280 12 415 0281 C9	SAVRO	HVI LDAX HVI STAX RET	E. ROWSOSO D E. ROWSAVE D	FOINT D-E TO BOBO ROWN FUT BOBO ROW N TO ACC FOINT D-E TO ROW SAVE STORE ROW SAVE N IN REF T RETURN
81 01E2.CA0502 82		JZ	SCROLL		416 417			U 5070 1005 SC	
83 84			MENT 6080 ROW #		418			W DATA LOAD RC	
85 01E5 3E2F 86 01E7 BE	INCRO	CMP	A. Ú2F M	, TEST FOR MAX ROW AND ; JUMP TO ZERO ROW IF TRUE	419 0282 1A 420 0283 5F	LDHL1	MŮV	D E. A	; LOAD ACC WITH D-E DATA ; POINT D-E TO N R. S DATA
87 01ES CAFB01 88 01EB 34		JZ INR	ZROW M	, ZERO ROW ; INCREMENT THE SOSO ROW #	421 0284 1A 422 0285 67		HOV	D H. A	ROW # TO N R S DATA HIGH
89 01EC EB		XCHG MVI	E. ROWSOSO	, POINT H-L TO CHAR #	423 0286 7B 424 0287 C630		ADI	A. E 030	PUT 1ST ROW # TO ACC
91 01EF CD8202		CALL	LEHL	RETURN	425 0289 5F 426 028A 1A		HOV	E. A D	FOINT D-E TO N R S. DATA
92 01F2 C9 93		RET		; RETURN	427 028B 6F 428 028C C9		MOV	LA	ROW # TO L REG
94 95			CHARACTER		429		LINEF		, ACTOMA
96 01F3 3E00 97 01F5 32633F	ZCHAR	STA	A, 000 03F63	, PUT CHAR # TO ZERO ; AND STORE	430 431				
98 01FS C3B301		JMF	PCUR	GO TO PUT CURSOR ROUTINE	432 028D CDDC01 433 0290 CDC401		CALL	NXRO1 CLROW3	OFF SCREEN CLEAR ROW ROUT
00		, ZERO 8	8080 ROW #		434 0293 1E61 435 0295 CD8202		CALL	E, ROWSOSO LEHL	; MOVE REFERENCE ROW * TO H ; LOAD H-L
02 01FB 3600 03 01FB 2E00	ZROW	MVI	M. 000 L. 000	;8080 ROW # TO ZERO ;N R S ADDRESS HIGH	436 0298 3A633F 437 029B 85	ADDCH	ADD	03F63 L	; CHAR # TO ACC ; ADD THE CHAR # TO THE
014 01FF 56		MOV	D. M	IN R. S. DATA HIGH TO D REG	438 029C 6F 439 029D 7C		MOV	L. A A. H	FIRST ROW ADDRESS. IF A CARRY OCCURED ADD TO
05 0200 2E30 06 0202 SE		MOV	E. M	IN. R. S DATA LOW TO E REG	440 029E CE00		ACI	000 H, A	THE DATA HIGH
07 0203 EB 08 0204 C9		XCHG RET		; EXCHANGE H-L WITH D-E ; RETURN	441 02A0 67 442 02A1 C3B301		JMP	PCUR	, PUT CURSOR TO LINE FED RO
09 10		; ROW SO	CROLL		443		HOME	CURSOR TO T. O	P.
1 12 0205 2B	SCROLL:	DCX	н	POINT H-L TO LAST ROW	445 446 02A4 1E62	HOME	HVI -	E.FIRSTRO	POINT D-E TO 1ST ROW
3 0206 3E2F 4 0208 BE		MVI	A, 02F	; BEFORE SCRATCH TABLES. ; TEST FOR THE LAST ROW.	447 02A6 1A 448 02A7 1E61		HVI	D E. ROHSOSO	STORE FIRSTROW TO ROW8030
15 0209 CA1902		JZ	ZLRO	JUMP TO ZERO LAST ROW IF TR'	449 02A9 12 450 02AA CD8302		STAX	D LDHL1	MOVE REFERENCE ROW TO H-L
16 020C 34		INR	М	, INCREMENT TO NEXT ROW.	451 02AD 3E00 452 02AF 32633F		MVI	A. 000 03F63	; PUT CHAR * BACK ; TO ZERO
19					453 02B2 C3B301		JMP	PCUR	PUT CURSOR HOME
20 020D 2E62 21 020F BE	ROLO:	CMP	L.FIRSTRO	; POINT M-L TO FIRST ROW# ; IS FIRST LOW = TO LAST ROW	454 455		; SHAP	ROWS	
22 0210 CA1E02 23 0213 34		JZ INR	ZFR0 M	JUMP TO ZERO FIRST R INCREMENT TO NEXT ROW	456 457 0285 1E65	SWAP:	HVI	E, ROWSAVE	FOINT D-E TO ROW SAVE . A
4 0214 2E61 5 0216 C3E501		MVI	L.ROW8080 INCRO	FOINT H-L TO 8080 ROW; GO TO INCREMENT ROW ROUTINE	458 02B7 CD8202 459 02BA 22663F		SHLD	LDHL 03F66	FUT IN H-L REG.
6		CHI	2140110	700 TO THE MENT NOW THE STATE	460 02BD 1E61 461 02BF CD8202		MVI	E, ROW8080	; POINT D-E TO 8080 ROW # A ; PUT ADDRESS IN H-L REG
7	71.00		M 000	DUT LACT DOUB TO ZEDO	462 02C2 1E65 463 02C4 1A		MVI	E. ROWSAVE	POINT D-E TO ROW SAVE # A
30 021B C30D02	ZLRO:	MVI JMP	M, 000 ROLO		454 02C5 5F .		MOV	E. A	: 8030 ROW # TO ADD HIGH
31		2.			465 02C6 7C 466 02C7 12	**	STAX	D D	; STORE 8080 ROW # TO N R. S ; DATA HIGH
33 021E 3600 34 0220 2E61	ZFRO	MVI	M. 000 L. ROW8080	; PUT FIRST ROW* TO ZERO ; POINT H-L TO 8080 ROW	467 02C8 7B 468 02C9 C630		ADI	A. E 030	
85 0222 C3E501		UMP	INCRU	GO TO INCREMENT ROW ROUTINE	469 02CB 5F 470 02CC 7D		MOV	E. A A. L	PUT 8080 ROW # TO N R. S. DATA LOW
37		; NEW RO	W START INTERR	UPT	471 02CD 12 472 02CE 2A663F		STAX	D 03F66	; 8080 ROW # IS NOW IN ROW ; PUT ROW SAVE # BACK TO H-
	NEWRO:	PUSH	PSW H	SAVE ACC AND FLAGS	473 02D1 1E61 474 02D3 1A		LDAX	E.ROH8080	COMENT SAME AS ABOVE
1 0227 D5		PUSH	D		475 02D4 5F 476 02D5 7C		MOV	E. A A. H	
12 0228 11643F 13 022B 1A		LDAX	D. 03F64 D	; POINT D-E TO CRTCROW # ; LOAD ACC WITH CRTC ROW #	477 02D6 12		STAX	D	
4 022C 5F		LDAX	E, A	; N. R. S. DATA ADD HIGH TO E ; ROW DATA HIGH INTO ACC	478 02D7 7B 479 02D8 C630		ADI	A, E 030	
6 022E C620 7 0230 67		MOV	H. A	IN. R. S. DATA ADD HIGH INTO H	490 OZDA 5F 481 OZDB 7D		MOV	E. A A. L	
8 0231 7B 9 0232 C630		MOV	A. E 030	ACC TO N R S DATA LOW	482 02DC 12 483 02DD C39802		STAX	D ADDCH	JUMP TO ADD CHAR
0 0234 5F 1 0235 1A		MOV	E. A D	IN R S. DATA ADD LOW TO E REG			BACK		
2 0236 6F		MOV	L, A M, 001	IN R. S. DATA ADD LOW INTO L.	486 487 02E0 1E63	BS:	MVI	E, CHARNUM	FOINT THE D-E REG TO CHAR
3 0237 3601 4 0239 D340		TUO	040	RESET N R S. AND VERT INTER	48S 02E2 1A		LDAX	D	, AND FUT IN ACC
5 023B 1E64 6 023D 1A		LDAX	E. CRTCROW D		499 02E3 FE00 490 02E5 CAEE02		CPI	UPROW	TEST FOR THE CHAR # = TO ZERO. JUMP IF TRUE
7 023E FE2F 8 0240 CA4A02		CF1 JZ	02F ZCRTC	; TEST FOR CRTC MAX ROW ; IF TRUE ZERO ACC	491 02E8 3D 492 02E9 12		STAX	A D	STORE DECREMENTED CHAR #
9 0243 3C	LOOP:	INE	A D	INCREMENT TO NEXT ROW STORE NEXT ROW NUMBER	493 02EA 2B 494 02EB C3B301		DCX JMP	H PCUR	; DEC H-L FOR NEW CURSOR LO ; PUT CURSOR IN DECREMENTED
1 0245 D1 2 0246 E1		FOP FOP	D H	RESTORE H-L REG	495 496		NEXT		
3 0247 F1 4 0243 FB		POP	FSW	RESTORE ACC AND FLAGS	497 498 02EE 3E4F	UPROW	MVI	A. 04F	; MOVE THE CHAR #
5 0249 C9		RET		FETURN	499 02F0 12 500	O. HOW	STAX	D	TO SOH AND STORE IT
7		7550 5	PTCPOU		501 502		, MOVE	CURSOR UP	
9	20020	; ZERO C		TERO ACC	503 02F1 EB	UPCUR.	XCHG	1 pougoon	FOINT H-L TO 8080 ROW AND
1 024C C34402	ZCRTC.	JMP	A. 000 LOOP	; ZERO ACC	504 02F2 2E61 1505 02F4 7E		MOV	L. ROW8080 A. M	, TO NEW CURSOR LOCATION ; TEST IF NEXT UP CURSOR WI
2 3		; VERTIC	CAL INTERRUPT		506 02F5 23 507 02F6 BE		CMP	H	FIF TRUE JUMP TO
	VERTI.	PUSH	PSW	SAVE ACC AND FLAGS	508 02F7 CA0803 509 02FA 2B		DC X	UPSCL H	POINT H-L BACK TO BOSO RO
6 0250 E5 7 0251 D5		PUSH	H D	SAVE H REG	510 511 02FB FE00	BACK1	CPI	000	IF 8080 ROW # IS EQUAL TO
9 0252 1E62 9 0254 1A		MVI	E. FIRSTRO	; POINT D-E TO FIRST ROW * ; LOAD 1ST ROW * INTO ACC	512 02FD CA1E03 513 0300 35		JZ DCR	RO4S M	JERO JUMP TO ROW 48 ROUTE DECREMENT 8080 ROW #
0 0255 1E64		MVI	E, CRTCROW	, POINT D-E TO CRICROW #	514	LOOP1	XCHG		
1 0257 12 2 0258 E63F		ANI	03F	REMOVE MARKER	515 0301 EB 516 0302 CD8202		CALL	LDHL	AND DIE TO BOBO ROW # JU
3 025A 5F 4 025B 1A		LEAX	E. A D	FOINT H L TO CRTC FIRST ROW	517 0305 C3°802 518		JMP	ADDCH	; TO ADD CHARACTER ROUTINE
5 025C C620 6 025E 67		MOV	H. A 40		519 0308 7E 520 0309 FE00	UPSCL.	MOV	A- M 000	, FUT FIRST ROW # INTO ACC. , TEST IF FIRST ROW # IS =
7 025F 7B 9 0260 C630		MOV	A. E 030		521 030B CA2403 522 030E 35		JZ DCR	FR04S	. ZERO IF TRUE JUMP TO ROW , 48 ROUTINE
9 0262 SF		MOV	E. A		523 524 030F 2E60	L00P2	MVI	L, LASTRON	
		MOV	L. A M. 002	STORE TOP OF PAGE	525 0311 7E 526 0312 FE00	20012	MOV	A. M 000	
0 0263 1A 1 0264 6F		TUO	040	, stone for the first	527 0314 CA2A03		JZ	LR048	
0 0263 1A 1 0264 6F 2 0265 3602 3 0267 0340					528 0317 35		DCR	M	
0 0263 1A 1 0264 6F 2 0265 3602 3 0267 0340 4 0269 01 5 026A E1		POP	D H		529				501117 / · · · · · · · · · · · · · · · · · ·
0 0263 1A 1 0264 6F 2 0265 3602 3 0267 0340 4 0269 D1		POP		RESTORE ACC AND FLAGS			MVI MOV JMP	L. ROWSOSO A. M BACK1	, POINT H-L TO SOSO ROH .

534	031F	3F2F	ROAR	PEUT	A. 02F	CHANGE BOBO ROW W
535	0320	77		MOV	M. A.	TO 23D AND STORE
536	0321	C30103		JMP	LOOP1	; CHANGE 8080 ROW * ; TO 23D AND STORE ; JUMP TO POINTER EXCHANGE ROW
5.37						
538	0324	3E 2F	FR048	HOV HOV	A. 02F	
539	0359	77		MOV	M. A	
540	0327	C30F03		JMF	LOOP2	
541	0220	25.25	15:048	MUT	A. 02F	FUT THE 1ST ROW TO
542	0326	77	2	MOV	M. A	i 17H
544	0320	C31803		JMP	A. 02F M. A LOOF3	17H >JUMP TO BOSO ROW # STORE
545				-		
546				CLEAR	ROW ROUTINE	
547						
548	0330	CD3903	CLROW		CLROW1	
549	0333	C 36F07		JITIP.	CR	
550	0224	1541	CL ROLLS	MUT	F. ROURORO	
221	0336	CD9202	CENOWI	COLL	I DHI	PUT ROW DATA IN H-L REG
553	0338	3E50	CL ROW2	MUI	A. 050	INTILIZE LOOP COUNTER.
55.4	0330	3620	1.0064	MVI	M. 020	STORE ASCII SPACE IN MEM.
555	033F	30		DCR	A	DECREMENT LOOP COUNTER.
556	0340	C8		RZ		FRETURN IF ZERO BIT IS SET.
557	0341	23		INX	- H	NEXT LOCATION
558	0342	C33D03		JMP	LOOP4	; PUT ROW DATA IN H-L REG ; INTILIZE LOOP COUNTER ; STORE ASCII SPACE IN MEM. ; DECEMENT LOOP COUNTER ; RETURN IF ZERO BIT IS SET. , NEXT LOCATION ; CLEAR NEXT LOCATION
						RING BELL
560	0345	D301	BELL	RET	001	INING DELL
562		C9		MEI		
543	0348	AF	IVERTN	XRA	A E. IMASK D	
564	0349	1E68		MVI	E, IMASK	POINT D.E TO MASK
565	034B	1A		LDAX	D	
566	034C	17		RAL		CK BIT 8 STATUS
567	034D	DA5203		JC	RESET	
568	0350	3E80		MVI	A. 080	INVERT BIT 8
569	0352	12	RESET.	STAX	D	STORE OUT NEW MASK
570	0353	C9		RET	ESET A. 080 D	
571					H E. ROWBOSO	
572	0354	F2	IVERTR.	PUSH	H POUGOGO	
5/3	0355	1501		COLL	E, KOWSOSO	LOAD 1ST ADD. OF BOBOROW TO
575	0357	1550		MUI	E. 050	SET COUNTER
576	0350	7F	LOOPA-	MOV	A. M	GET CHAR.
577	035D	17		RAL	E. ROWSOSO LDHL E. 050 A. M	CK BIT 8 STATUS AND INVERT
578	035E	DA7003		JC	RESET1	
579	0361	1F		RAR		
580	0362	F680		ORI	080	MASK BIT 8 HIGH
581	0364	77	BACK2:	ORI MOV INX MOV CPI JZ DCR JMP	M. A	STORE MOD. CHAR TO MEM
582	0365	23		INX	H	POINT TO NEXT MEM
583	0366	7B		MOV	A, E	
584	0367	PEUI		CPI	001	RETURN IF COUNT = ZERO
585	0369	LA7603		000	E	DEC. COUNTER
507	0360	U32003		IMP	LOOP6	PEC. COONTEN
588	0300	030003		0.0	20010	
589	0370	1F	RESET1.	RAR		
590	0371	E67F	RESET1.	ANI	07F	RESET BIT 8
591	0373	C36403			BACK2	
592						
593	0376	El	DONE.	FOP	Н	
374	03//	Cy		NE I		
595		0000		. END	START	

A	0007	ACELD	0110	ADCUR	0161	ADDCH	0298
B	0000	E110	00D4	B1200	OOEC	B150	CODA
B1800	00F2	B2000	00F8	B2400	OOFE	B300	00E0
B3600	0104	P4800	010A	E600	OOEA	B7200	0110
B9600	0116	BACK	0083	BACK1	02FB	BACK 2	0364
PAUD	0093	BELL	0345	BS	02E0	C	0001
CHARNU	0063	CLRAM	0042	CLRAM1	004C	CLROW	0330
CLROW1	0336	CLROW2	033B	CLROW3	0104	CR	026E
CRICRO	0064	D	0002	DONE	0376	E	0003
FIRSTR	0062	FR048	0324	FUNC	0170	H	0004
HMCUR	0087	HOME	02A4	IMASK	0068	INCRO	01E5
INIT	003B	INTACE	014A	INTER	0136	IVERTN	0348
IVERTR	0354	L	0005	LASTRO	0060	LDHL	0282
LDHL1	0283	LF	0280	LOOP	0244	LOOP1	0301
LOOP2	030F	LOOF3	0318	LOOP4	033D	LOOPS	OICE
LOOPE	0350	LR043	032A	m	0006	NEWRO	0225.
NRS	0061	NXRO	OIBE	NXR01	OIDC	FCUR	01B3
PSW	0006	RESET	0352	RESET1	0370	R049	031E
ROLO	020D	ROWSOS	0061	ROWSAV	0065	ROZERO	0107
SAVRO	027B	SCROLL	0205	SP	0006	START	0000
SWAP	0285	TEMP1	0066 *	TEMP2	0067 *	UFCUR	02F1
LIPRON	02EE	UPSCL	0308	VERTI	024F	ZCHAR	01F3
			021E	ZLRO	0219	ZROW	DIFE
ZCRTC	024A	ZFRO	021E	ZLRO	0219	ZROW	OIF

NO ERROR LINES
SOURCE CHECKSUM = 403F
OBJECT CHECKSUM = 0F51
INPUT FILE 1 CRTSOA SRC ON JIMFM
OBJECT FILE 1 CRTSOA LM ON JIMFM

DEFINITIONS

ACE - Asynchronous communication element CRTC - Cathode ray tube controller Video Page - Visible screen data Video RAM - Entire portion of RAM used only for display First Row # - Address for top row of video page Last Row # - Address for bottom row of video page CRTC Row # - Address for next row load 8080 Row # - Address for cursor row Character # - Character location in a row XXXH are hexidecimal numbers

REFERENCES

National Semiconductor Data Sheets:

DP8350 Series Programmable CRT Controllers **INS8250 Asynchronous Communications Element** DM8678 Bipolar Character Generator INS8080 Assembly and Reference Manuals

National Semiconductor Application Notes:

Simplify CRT Terminal Design with the DP8350, AN-198

DM8678 Bipolar Character Generator, AN-167

Data Bus and Differential Line Drivers and Receivers, AN-83

Transmission Line Characteristics, AN-108

Hardware Reference Manual BLC 80/10 Board Level Computer. National Semiconductor Microcomputer Systems Chapter 6 - System Interfacing.



National Semiconductor Corporation 2900 Semiconductor Drive Santa Clara California 95051 Tel (408: 737-500) TWX (910) 339-9240

National Semiconductor 6mbH 8000 Munchen 21 Hisenheimerstrasse 61/2 Hisenheimerstrasse 61/2 High 9701suya Shinjuku-k Tokyo Japan Tel 089 9 15027 Hel (03) 355-3711 Telex 05-22772

NS International Inc., Japan Miyake Building 1-- 9 Yotsuya Shinjuku-ku 160 Tokyo Japan Tel (03) 355-3711 TWX 232-2015 NSCJ-J

National Semiconductor [Hong Kong] Ltd
8th Floor
Cheung Kung Electronic Bldg
4 Hing Yip Street
Kwun Toing
Kowloon Hong Kong
Tel 3 411/241.8
Telex 73866 NSFHK HY Teles 73866 NSEHK HX Cable NATSEMI

NS Electronics Do Brasil
Avda Brigadeiro Faria Lima 844
11 Andar Conjunto 1104
12 Ardim Paulistano
Sao Paulo Brasil
Telex 1121003 CABINE SAO PAULO

NS Electronics Pty Ltd
Cnr Stud Rd & Min. Highway
Bayswater Victoria 3153
Australia
Tel 03-729-6333
Telex 122003 CABINE SAO PAULO
Telex 320%5

